Detect_fraudulent_Auto_Insurance_claims_extended

May 10, 2022

1 Auto Insurance Claims Fraud Detection

[ ]:

[ ]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

2 Business Requirement

An insurance company has approached you with a dataset of previous claims of their clients. The insurance company wants you to develop a model to help them predict which claims look fraudulent. By doing so you hope to save the company millions of dollars annually.

Claim related fraud is a huge problem in the insurance industry. It is quite complex and difficult to identify those unwanted claims. With Random Forest Non-Parametric Machine Learning Algorithm, I am trying to troubleshoot and help the General Insurance industry with this problem.

The data that I have is from Automobile Insurance. I will be creating a predictive model that predicts if an insurance claim is fraudulent or not. The answere between YES/NO, is a Binary Classification task. A comparison study has been performed to understand which ML algorithm suits best to the dataset.

[ ]:
```
import os
os.getcwd()
```

[ ]:
```
'/content'
```

[ ]:
```
#Importing required libraries

import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
from sklearn.ensemble import ExtraTreesRegressor
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.model_selection import train_test_split
```

```
import numpy as np
import sklearn.metrics
from pylab import rcParams
%matplotlib inline
pd.set_option('display.max_columns', 500)
pd.set_option('display.max_rows', 500)

# pandas version 0.24 or upper is required
pd.__version__
```

[ ]:
```
'1.1.5'
```

[ ]:
```
#load & view raw data
df = pd.read_csv('/content/drive/My Drive/Colab Notebooks/
 ↪insurance_claim_updated.csv')
df.head(10)
```

[ ]:

	months_as_customer	age	policy_number	policy_bind_date	policy_state
0	5	37	939011	16-07-2002	IN
1	462	58	902576	28-11-2002	IL
2	198	51	575784	12-05-2007	OH
3	384	47	102488	10-02-1998	OH
4	100	27	1129102	01-03-2012	IL
5	306	50	769636	09-01-1999	IN
6	105	27	948899	04-04-2003	IL
7	249	49	330251	24-07-2012	IN
8	44	28	136414	08-05-1995	IL
9	104	35	701032	05-07-2014	IL

	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit
0	250/500	500	1145.28	0.0
1	500/1000	1000	1156.80	0.0
2	100/300	2000	751.02	0.0
3	100/300	500	1137.34	1000000.0
4	100/300	2000	1082.70	4000000.0
5	100/300	500	1386.90	3000000.0
6	250/500	2000	1161.78	5000000.0
7	100/300	2000	1411.67	0.0
8	100/300	500	1529.81	0.0
9	250/500	500	1233.94	0.0

	insured_zip	insured_sex	insured_education_level	insured_occupation
0	360963	FEMALE	Associate	priv-house-serv
1	432568	FEMALE	MD	exec-managerial
2	712296	FEMALE	High School	farming-fishing
3	402197	FEMALE	High School	transport-moving
4	577005	FEMALE	PhD	armed-forces

```
5       478119      FEMALE             High School       armed-forces
6       377775        MALE                       MD    farming-fishing
7       505288        MALE                       JD   transport-moving
8       436652        MALE                 Associate       tech-support
9       538379        MALE                 Associate    protective-serv

  insured_hobbies insured_relationship  capital.gains  capital.loss  \
0          movies               husband          54735         88553
1         camping         other-relative           1381         50621
2            golf              own-child              0             0
3  bungie-jumping               husband              0         42211
4        exercise               husband              0             0
5         dancing                  wife              0          5905
6      basketball          not-in-family              0          1185
7          hiking              own-child          16759         77869
8        kayaking          not-in-family              0             0
9          movies               husband              0         19249

  incident_date             incident_type   collision_type incident_severity  \
0    06-02-2015  Single Vehicle Collision  Front Collision      Minor Damage
1    18-01-2015   Multi-vehicle Collision   Rear Collision        Total Loss
2    13-02-2015               Parked Car                ?    Trivial Damage
3    27-01-2015            Vehicle Theft                ?    Trivial Damage
4    21-02-2015            Vehicle Theft                ?      Minor Damage
5    17-02-2015  Single Vehicle Collision   Rear Collision      Minor Damage
6    12-02-2015   Multi-vehicle Collision   Rear Collision        Total Loss
7    17-02-2015  Single Vehicle Collision  Front Collision        Total Loss
8    03-01-2015  Single Vehicle Collision   Rear Collision      Minor Damage
9    21-02-2015   Multi-vehicle Collision   Side Collision      Major Damage

  authorities_contacted incident_state incident_city   incident_location  \
0                 Other             NY     Hillsdale         6770 1st St
1                 Other             SC     Arlington        3275 Pine St
2                  None             SC     Arlington     1741 Best Ridge
3                Police             WV   Springfield    9744 Texas Drive
4                  None             OH    Northbrook  3289 Britain Drive
5                  Fire             NY      Columbus       4633 5th Lane
6                 Other             VA    Northbrook      3653 Elm Drive
7                  Fire             NY     Riverwood     1580 Maple Lane
8             Ambulance             WV      Columbus    4862 Lincoln Hwy
9                 Other             NC     Arlington        7609 Rock St

   incident_hour_of_the_day  number_of_vehicles_involved property_damage  \
0                        20                            1               ?
1                        11                            2               ?
2                         0                            1              NO
3                         6                            1             YES
```

```
4                        5                             1              NO
5                        6                             2             YES
6                       11                             3              ?
7                        3                             1              NO
8                       22                             1              NO
9                       16                             4             YES

   bodily_injuries  witnesses police_report_available  total_claim_amount  \
0                2          1                      YES             96200.0
1                0          5                        ?             31200.0
2                1          3                        ?             14500.0
3                1          1                       NO              7500.0
4                2          1                      YES             16500.0
5                1          1                       NO             37710.0
6                0          1                      YES             36400.0
7                0          3                        ?             83000.0
8                3          2                        ?             34510.0
9                1          2                        ?             91100.0

   injury_claim  property_claim  vehicle_claim  auto_make  auto_model  \
0          3000             500          58870       Saab         92x
1          3830            7370          32130       Saab          95
2             0               0           5690  Chevrolet      Malibu
3             0               0            420       Saab          95
4          5400            4300           8270     Toyota  Highlander
5         14780            5400          48800      Honda       Civic
6          6100             190          27630  Chevrolet       Tahoe
7         16000             770          58000     Accura         RSX
8         11000            5880          24800        BMW          M5
9         11410           14160          59520       Audi          A5

   auto_year fraud_reported
0       1997              N
1       2006              N
2       1996              N
3       1990              N
4       1998              N
5       1995              N
6       2005              N
7       2009              N
8       2005              N
9       2000              N
```

[ ]:
```
df.describe()
```

[ ]:
```
       months_as_customer           age  policy_number  policy_deductable  \
count        10211.000000  10211.000000   1.021100e+04       10211.000000
```

mean	213.467927	39.050142	5.474680e+05	1159.044168
std	133.639732	11.508964	3.034069e+05	621.773731
min	0.000000	2.000000	4.410000e+02	500.000000
25%	106.000000	31.000000	3.095050e+05	500.000000
50%	202.000000	38.000000	5.364750e+05	1000.000000
75%	303.000000	47.000000	7.717955e+05	2000.000000
max	747.000000	79.000000	1.615353e+06	2000.000000

	policy_annual_premium	umbrella_limit	insured_zip	capital.gains
count	10211.000000	1.021100e+04	10211.000000	10211.000000
mean	1257.794204	1.727157e+06	503057.770248	16459.434531
std	300.874661	2.407828e+06	88766.415575	24596.437735
min	179.790000	0.000000e+00	203707.000000	0.000000
25%	1058.790000	0.000000e+00	436706.000000	0.000000
50%	1255.210000	1.000000e+06	485493.000000	155.000000
75%	1454.130000	3.000000e+06	569575.500000	26191.000000
max	2390.510000	1.000000e+07	788825.000000	134607.000000

	capital.loss	incident_hour_of_the_day	number_of_vehicles_involved
count	10211.000000	10211.000000	10211.000000
mean	17150.482029	11.229850	2.010087
std	25528.629014	6.411803	1.101773
min	0.000000	0.000000	1.000000
25%	0.000000	6.000000	1.000000
50%	0.000000	12.000000	2.000000
75%	27564.000000	16.000000	3.000000
max	142321.000000	24.000000	6.000000

	bodily_injuries	witnesses	total_claim_amount	injury_claim
count	10211.000000	10211.000000	10211.000000	10211.000000
mean	1.139262	1.652434	56608.934482	7912.681422
std	0.896285	1.195957	27647.190092	5456.281497
min	0.000000	0.000000	100.000000	0.000000
25%	0.000000	1.000000	37930.000000	3615.000000
50%	1.000000	2.000000	58200.000000	7460.000000
75%	2.000000	3.000000	76000.000000	11800.000000
max	4.000000	6.000000	154740.000000	30000.000000

	property_claim	vehicle_claim	auto_year
count	10211.000000	10211.000000	10211.000000
mean	8028.269513	40822.630497	2004.358927
std	5514.767560	19666.958809	6.442418
min	0.000000	10.000000	1981.000000
25%	3730.000000	27700.000000	2000.000000
50%	7600.000000	42200.000000	2005.000000
75%	11770.000000	54700.000000	2009.000000
max	29700.000000	110800.000000	2015.000000

[ ]:
```
df.dtypes
```

[ ]:
```
months_as_customer              int64
age                             int64
policy_number                   int64
policy_bind_date               object
policy_state                   object
policy_csl                     object
policy_deductable               int64
policy_annual_premium         float64
umbrella_limit                float64
insured_zip                     int64
insured_sex                    object
insured_education_level        object
insured_occupation             object
insured_hobbies                object
insured_relationship           object
capital.gains                   int64
capital.loss                    int64
incident_date                  object
incident_type                  object
collision_type                 object
incident_severity              object
authorities_contacted          object
incident_state                 object
incident_city                  object
incident_location              object
incident_hour_of_the_day        int64
number_of_vehicles_involved     int64
property_damage                object
bodily_injuries                 int64
witnesses                       int64
police_report_available        object
total_claim_amount            float64
injury_claim                    int64
property_claim                  int64
vehicle_claim                   int64
auto_make                      object
auto_model                     object
auto_year                       int64
fraud_reported                 object
dtype: object
```

[ ]:
```
df.columns
```

[ ]:
```
Index(['months_as_customer', 'age', 'policy_number', 'policy_bind_date',
       'policy_state', 'policy_csl', 'policy_deductable',
```

```
       'policy_annual_premium', 'umbrella_limit', 'insured_zip', 'insured_sex',
       'insured_education_level', 'insured_occupation', 'insured_hobbies',
       'insured_relationship', 'capital.gains', 'capital.loss',
       'incident_date', 'incident_type', 'collision_type', 'incident_severity',
       'authorities_contacted', 'incident_state', 'incident_city',
       'incident_location', 'incident_hour_of_the_day',
       'number_of_vehicles_involved', 'property_damage', 'bodily_injuries',
       'witnesses', 'police_report_available', 'total_claim_amount',
       'injury_claim', 'property_claim', 'vehicle_claim', 'auto_make',
       'auto_model', 'auto_year', 'fraud_reported'],
      dtype='object')
```

[ ]:
```
df.shape
```

[ ]:
```
(10211, 39)
```

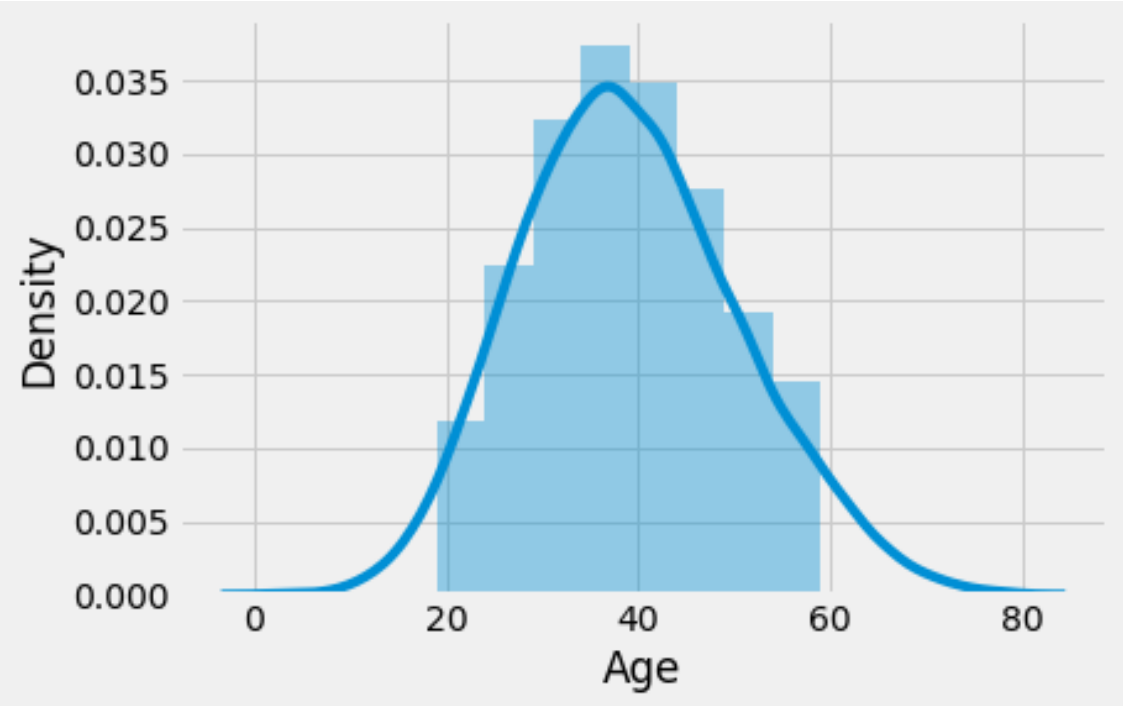
[ ]:
```
df.nunique()
```

[ ]:
```
months_as_customer             614
age                             76
policy_number                10163
policy_bind_date               951
policy_state                     3
policy_csl                       3
policy_deductable                3
policy_annual_premium         9706
umbrella_limit                  11
insured_zip                  10045
insured_sex                      2
insured_education_level          7
insured_occupation              14
insured_hobbies                 20
insured_relationship             6
capital.gains                 4940
capital.loss                  4871
incident_date                   60
incident_type                    4
collision_type                   4
incident_severity                4
authorities_contacted            5
incident_state                   7
incident_city                    7
incident_location             1000
incident_hour_of_the_day        25
number_of_vehicles_involved      6
property_damage                  3
bodily_injuries                  5
```

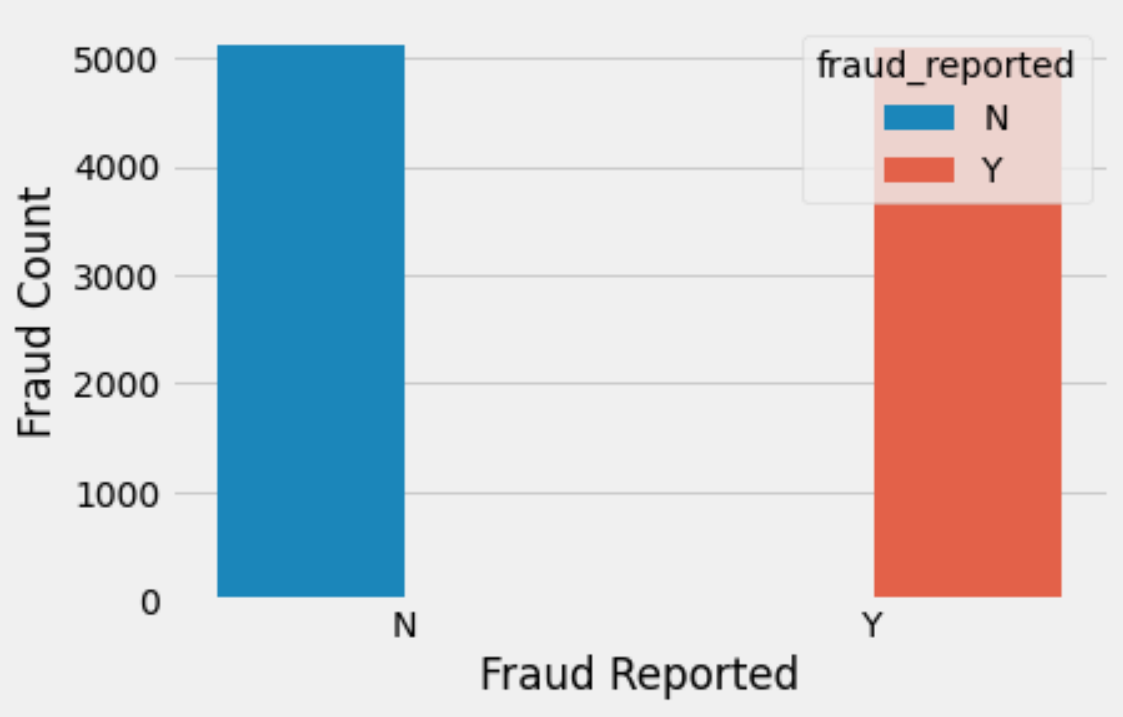
```
witnesses                      7
police_report_available        3
total_claim_amount          4816
injury_claim                1825
property_claim              1858
vehicle_claim               4147
auto_make                     14
auto_model                    39
auto_year                     35
fraud_reported                 2
dtype: int64
```

[ ]:
```
plt.style.use('fivethirtyeight')
ax = sns.distplot(df.age, bins=np.arange(19,64,5))
ax.set_ylabel('Density')
ax.set_xlabel('Age')
plt.show()
```

```
/usr/local/lib/python3.6/dist-packages/seaborn/distributions.py:2551:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `histplot` (an axes-level function for
histograms).
  warnings.warn(msg, FutureWarning)
```



[ ]:
```
plt.style.use('fivethirtyeight')
ax = sns.countplot(x='fraud_reported', data=df, hue='fraud_reported')
ax.set_xlabel('Fraud Reported')
ax.set_ylabel('Fraud Count')
plt.show()
```



From above plot, like most fraud datasets, the label distribution is skewed.

[ ]:
```
df['fraud_reported'].value_counts() # Count number of frauds vs non-frauds
```

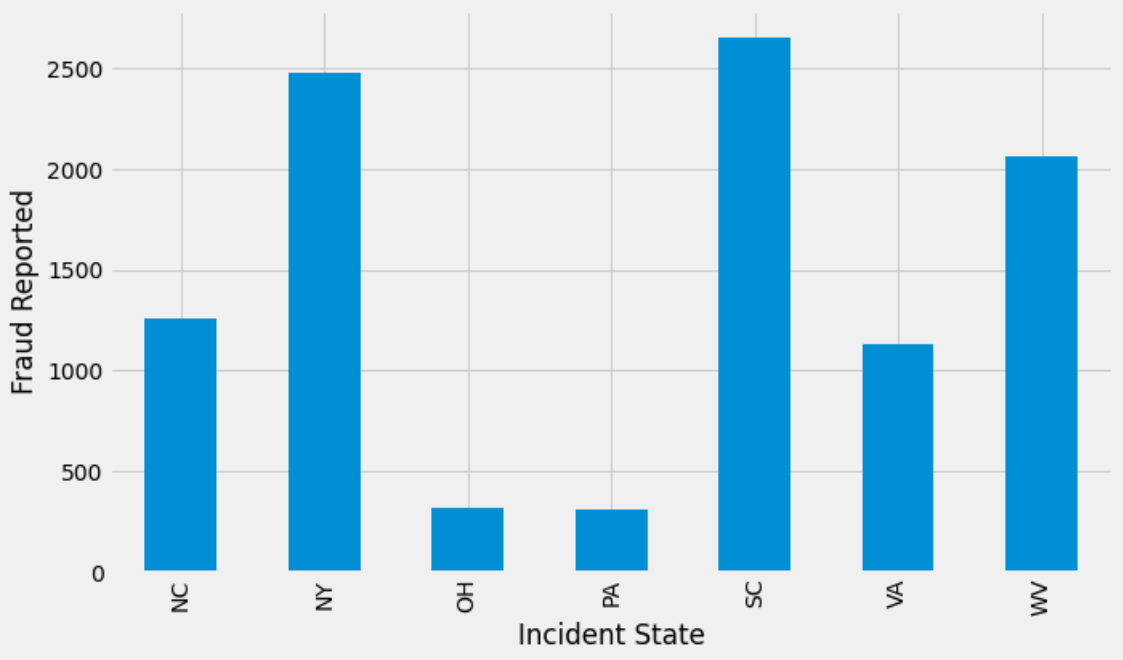
[ ]:
```
N    5118
Y    5093
Name: fraud_reported, dtype: int64
```

[ ]:
```
df['incident_state'].value_counts()
```

[ ]:
```
SC    2656
NY    2476
WV    2061
NC    1256
VA    1129
OH     321
PA     312
Name: incident_state, dtype: int64
```

Here we see that almost 25% fraud reported. Let's try to look for an indicative variable. Let's analyze location. This dataset only has information from the mid-Atlantic states from the USA.

[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(10,6))
ax = df.groupby('incident_state').fraud_reported.count().plot.bar(ylim=0)
ax.set_ylabel('Fraud Reported')
ax.set_xlabel('Incident State')
plt.show()
```



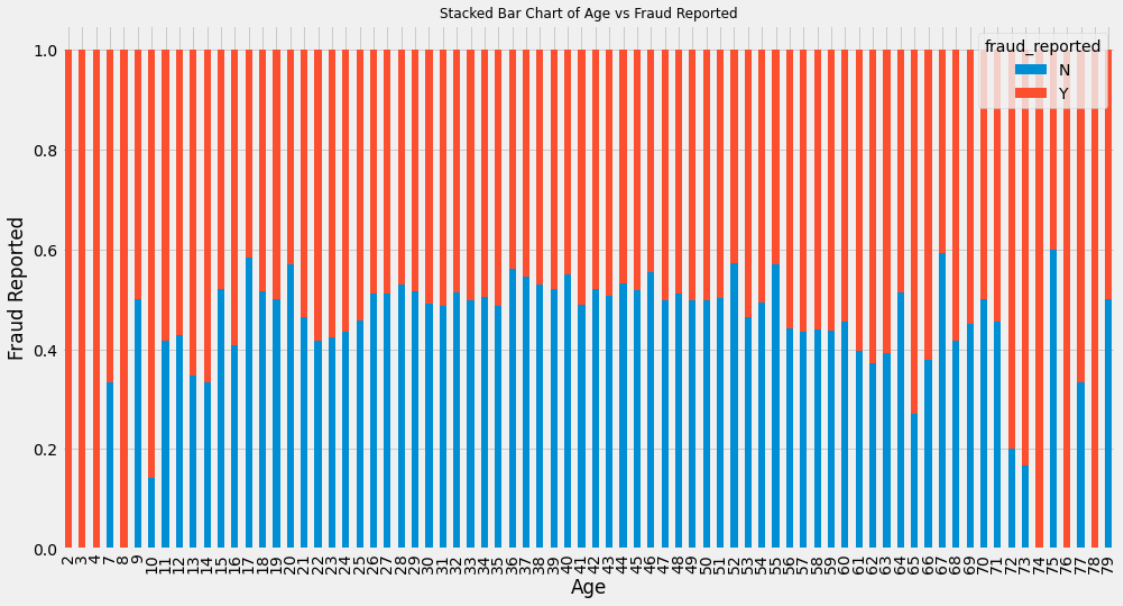
[ ]:
```
df['incident_state'].unique()
```

[ ]:
```
array(['NY', 'SC', 'WV', 'OH', 'VA', 'NC', 'PA'], dtype=object)
```

[ ]:
```
df['incident_state_count'] = df['incident_state']
for i in range(len(df['incident_state_count'])):
    if df.iloc[i, 39] == "NY":
        df.iloc[i, 39] = 262
    if df.iloc[i, 39] == "SC":
        df.iloc[i, 39] = 248
    if df.iloc[i, 39] == "WV":
        df.iloc[i, 39] = 217
    if df.iloc[i, 39] == "VA":
        df.iloc[i, 39] = 110
    if df.iloc[i, 39] == "NC":
        df.iloc[i, 39] = 110
```

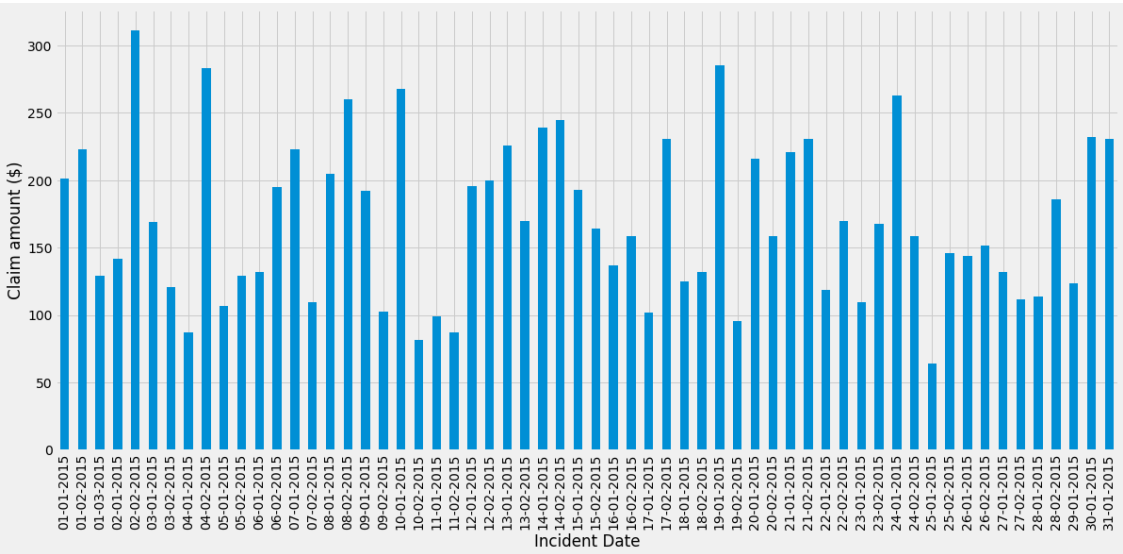
```
    if df.iloc[i, 39] == "PA":
        df.iloc[i, 39] = 30
    if df.iloc[i, 39] == "OH":
        df.iloc[i, 39] = 23

from plotly.offline import plot
import plotly.graph_objs as go

data = [go.Choropleth(autocolorscale = True, locations = df['incident_state'],
                      z = df['incident_state_count'],
                      locationmode = 'USA-states',
                      marker = go.choropleth.Marker(line = go.choropleth.marker.
 ↪Line(color = 'rgb(255,255,255)', width = 2)),
                      colorbar = go.choropleth.ColorBar(title = "Number of␣
 ↪Incidents"))]
layout = go.Layout(
    title = go.layout.Title(
        text = 'Insurance Incident Claims on the Mid-Atlantic'
    ),
    geo = go.layout.Geo(
        scope = 'usa',
        projection = go.layout.geo.Projection(type = 'albers usa'),
        showlakes = True,
        lakecolor = 'rgb(255, 255, 255)'),
)
fig = go.Figure(data = data, layout = layout)

#plot(fig, filename = 'd3-cloropleth-map')  # for showing in seprate tab
fig.show()
```

[ ]:
```
plt.rcParams['figure.figsize'] = [15, 8]
ax= plt.style.use('fivethirtyeight')
table=pd.crosstab(df.age, df.fraud_reported)
table.div(table.sum(1).astype(float), axis=0).plot(kind='bar', stacked=True)
plt.title('Stacked Bar Chart of Age vs Fraud Reported', fontsize=12)
plt.xlabel('Age')
plt.ylabel('Fraud Reported')
plt.show()
```



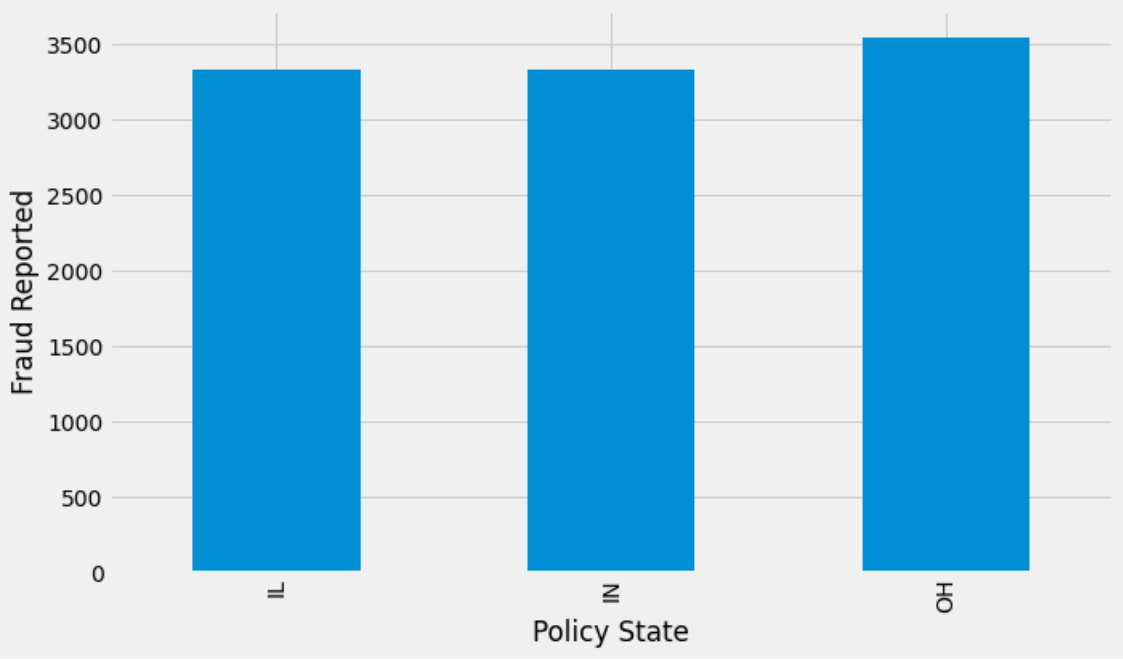
From above plot, it is obvious that, age is an important predictor for fraud reported. Age between 19-23 shows substantial number od fraud report.

[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(18,8))
ax = df.groupby('incident_date').total_claim_amount.count().plot.bar(ylim=0)
ax.set_ylabel('Claim amount ($)')
ax.set_xlabel('Incident Date')
plt.show()
```

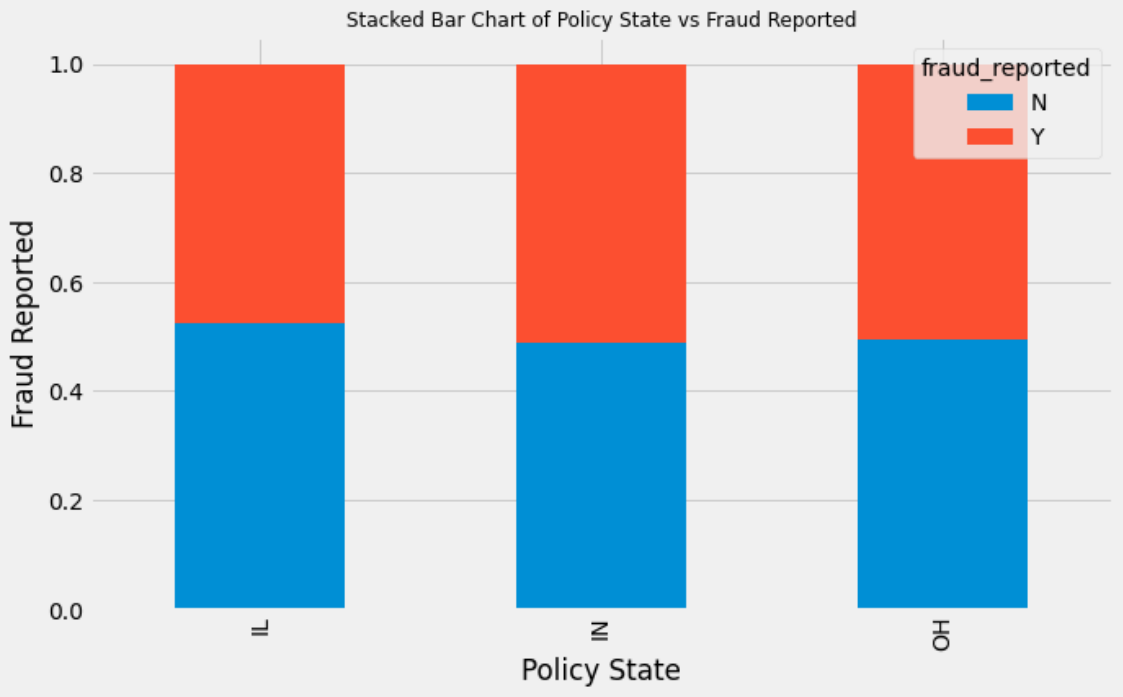


We see that, all the cases in above plot are for the months of January and February 2015

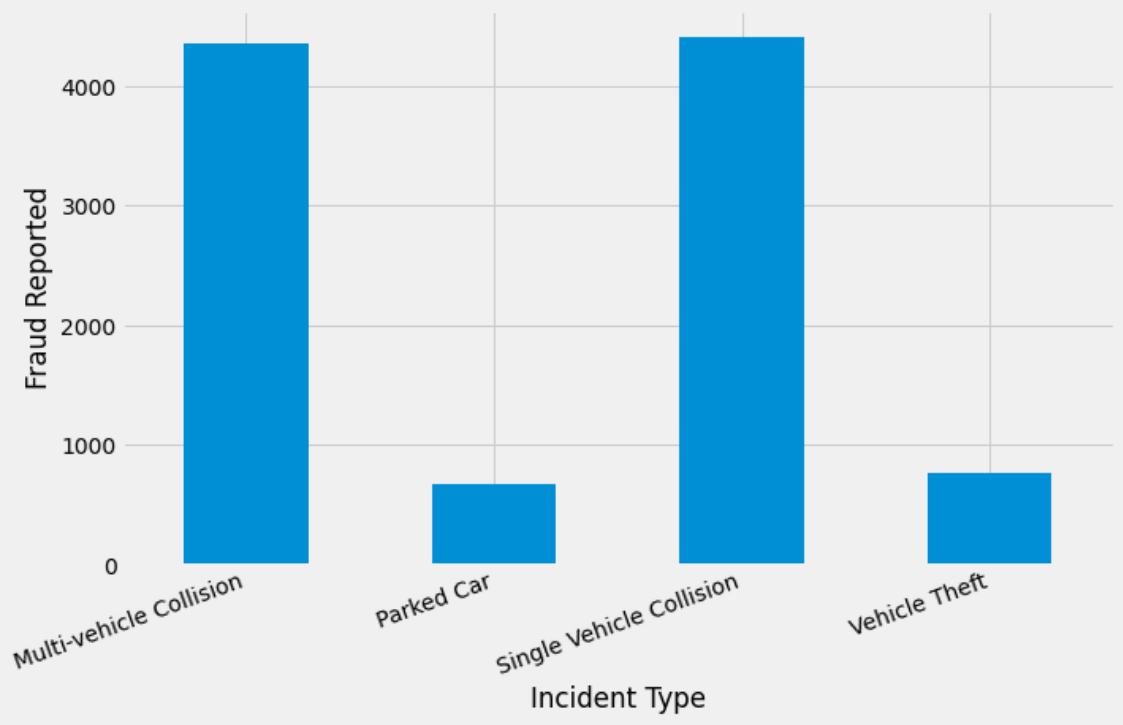
[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(10,6))
ax = df.groupby('policy_state').fraud_reported.count().plot.bar(ylim=0)
ax.set_ylabel('Fraud Reported')
ax.set_xlabel('Policy State')
plt.show()
```



[ ]:
```
plt.rcParams['figure.figsize'] = [10, 6]
ax= plt.style.use('fivethirtyeight')
table=pd.crosstab(df.policy_state, df.fraud_reported)
table.div(table.sum(1).astype(float), axis=0).plot(kind='bar', stacked=True)
plt.title('Stacked Bar Chart of Policy State vs Fraud Reported', fontsize=12)
plt.xlabel('Policy State')
plt.ylabel('Fraud Reported')
plt.show()
```

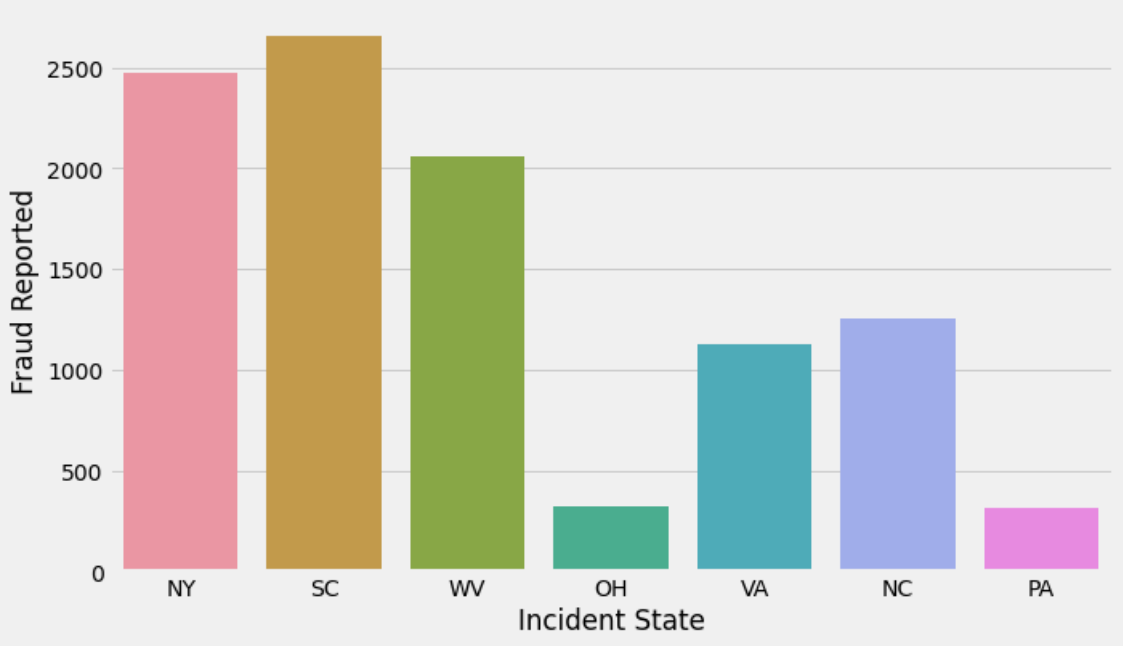


[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(10,6))
ax = df.groupby('incident_type').fraud_reported.count().plot.bar(ylim=0)
ax.set_xticklabels(ax.get_xticklabels(), rotation=20, ha="right")
ax.set_ylabel('Fraud Reported')
ax.set_xlabel('Incident Type')
plt.show()
```

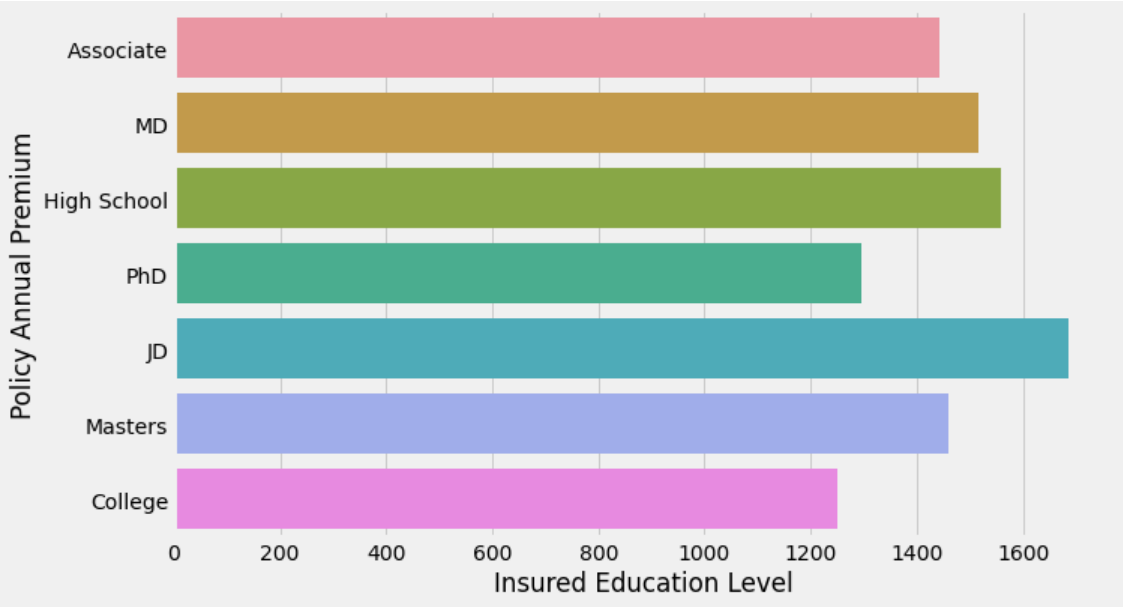


[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(10,6))
ax = sns.countplot(x='incident_state', data=df)
ax.set_ylabel('Fraud Reported')
ax.set_xlabel('Incident State')
```

[ ]:
```
Text(0.5, 0, 'Incident State')
```

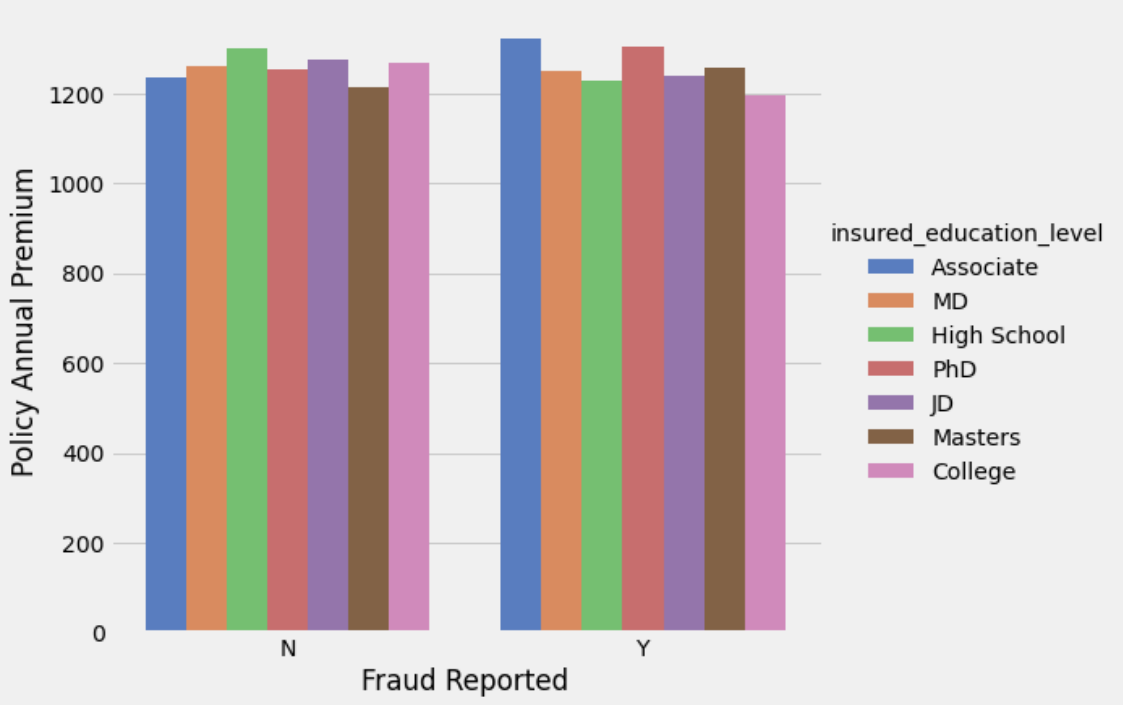


[ ]:
```
fig = plt.figure(figsize=(10,6))
ax = sns.countplot(y = 'insured_education_level', data=df)
ax.set_ylabel('Policy Annual Premium')
ax.set_xlabel('Insured Education Level')
plt.show()

# # Breakdown of Average Vehicle claim by insured's education level, grouped by␣
 ↪fraud reported
```

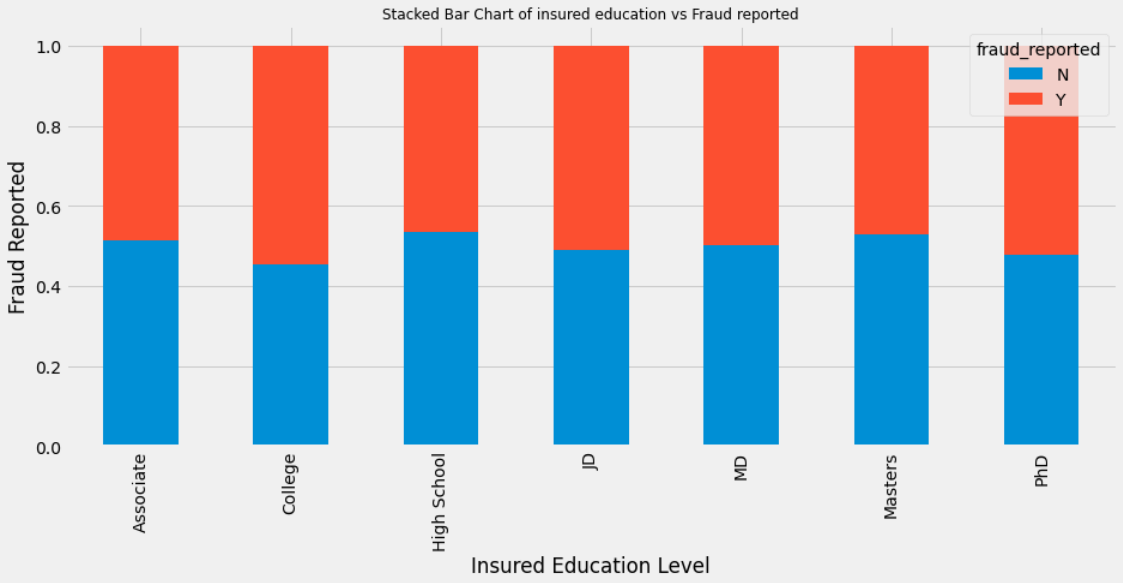


[ ]:
```
fig = plt.figure(figsize=(16,10))
ax = sns.catplot(x='fraud_reported',␣
 ↪y='policy_annual_premium',hue='insured_education_level', data=df,
                    kind="bar", ci=None, palette="muted",height=6, legend=True,␣
 ↪aspect=1.2)

ax.set_axis_labels("Fraud Reported", "Policy Annual Premium")

plt.show()
```

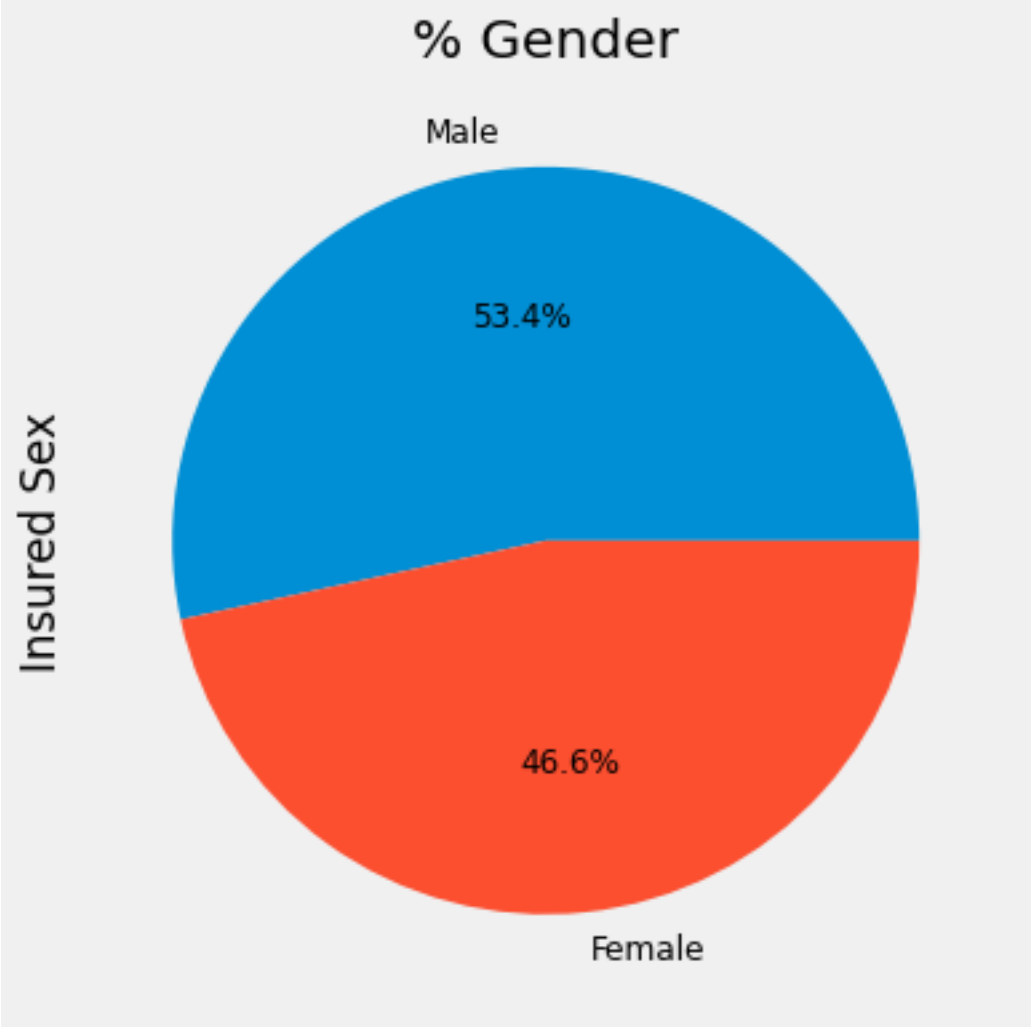
```
<Figure size 1152x720 with 0 Axes>
```



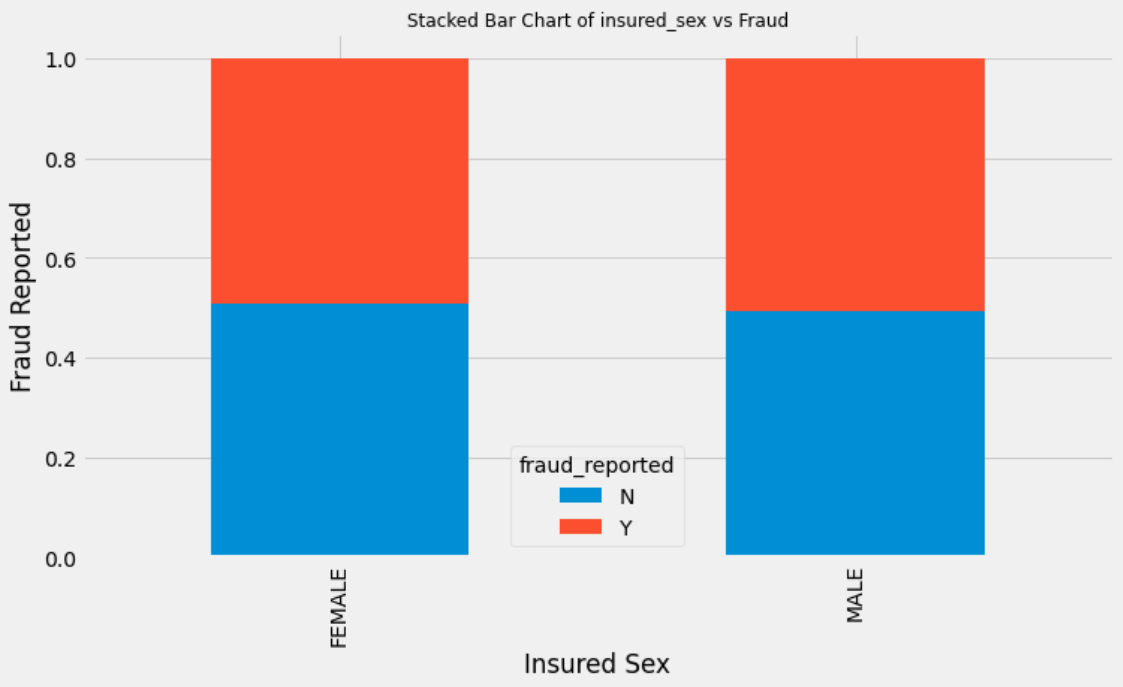
[ ]:
```
plt.rcParams['figure.figsize'] = [14, 6]
table=pd.crosstab(df.insured_education_level, df.fraud_reported)
table.div(table.sum(1).astype(float), axis=0).plot(kind='bar', stacked=True)
plt.title('Stacked Bar Chart of insured education vs Fraud reported',␣
 ↪fontsize=12)
plt.xlabel('Insured Education Level')
plt.ylabel('Fraud Reported');
```



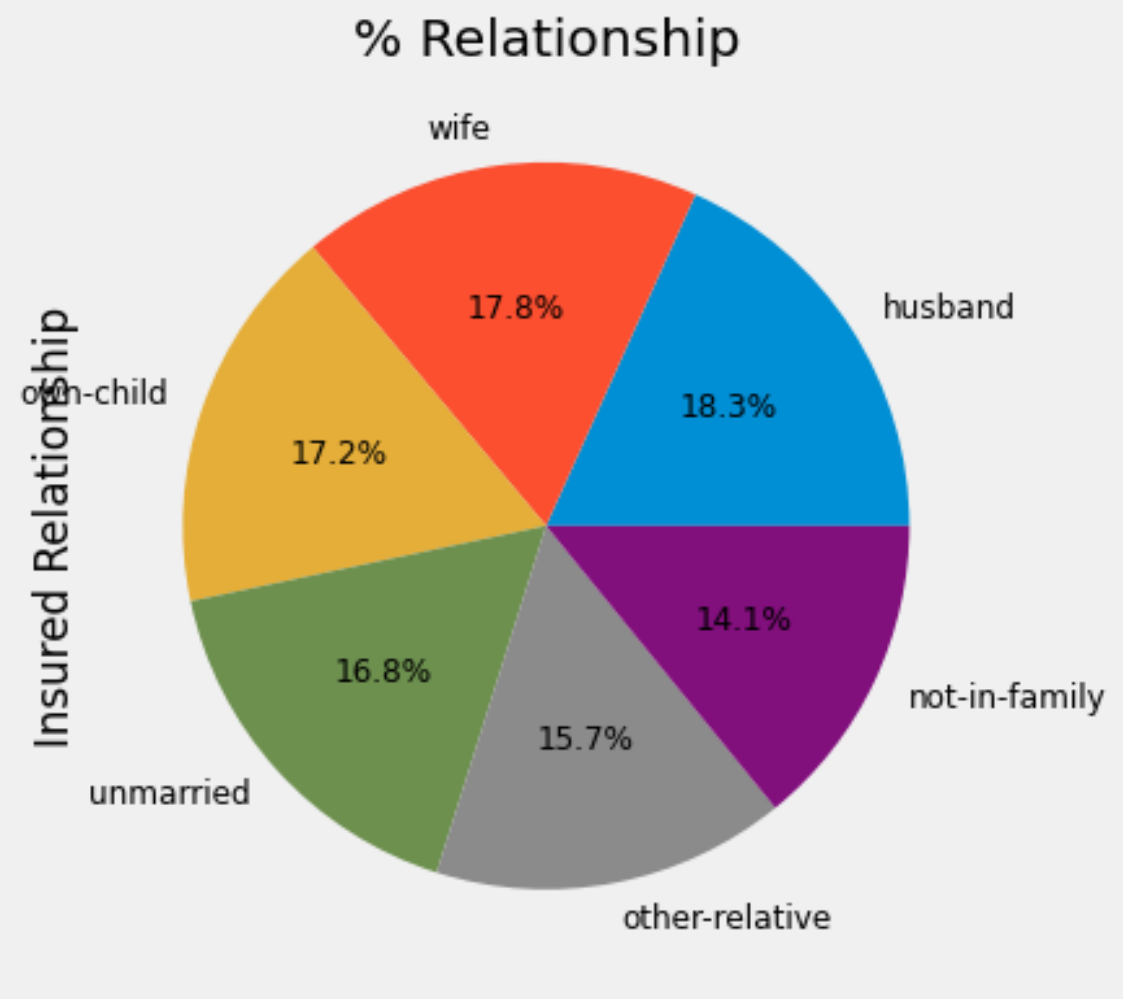
[ ]:
```
plt.rcParams['figure.figsize'] = [6, 6]
ax = (df['insured_sex'].value_counts()*100.0 /len(df))\
.plot.pie(autopct='%.1f%%', labels = ['Male', 'Female'], fontsize=12)
ax.set_title('% Gender')
plt.ylabel('Insured Sex')
plt.show()
```



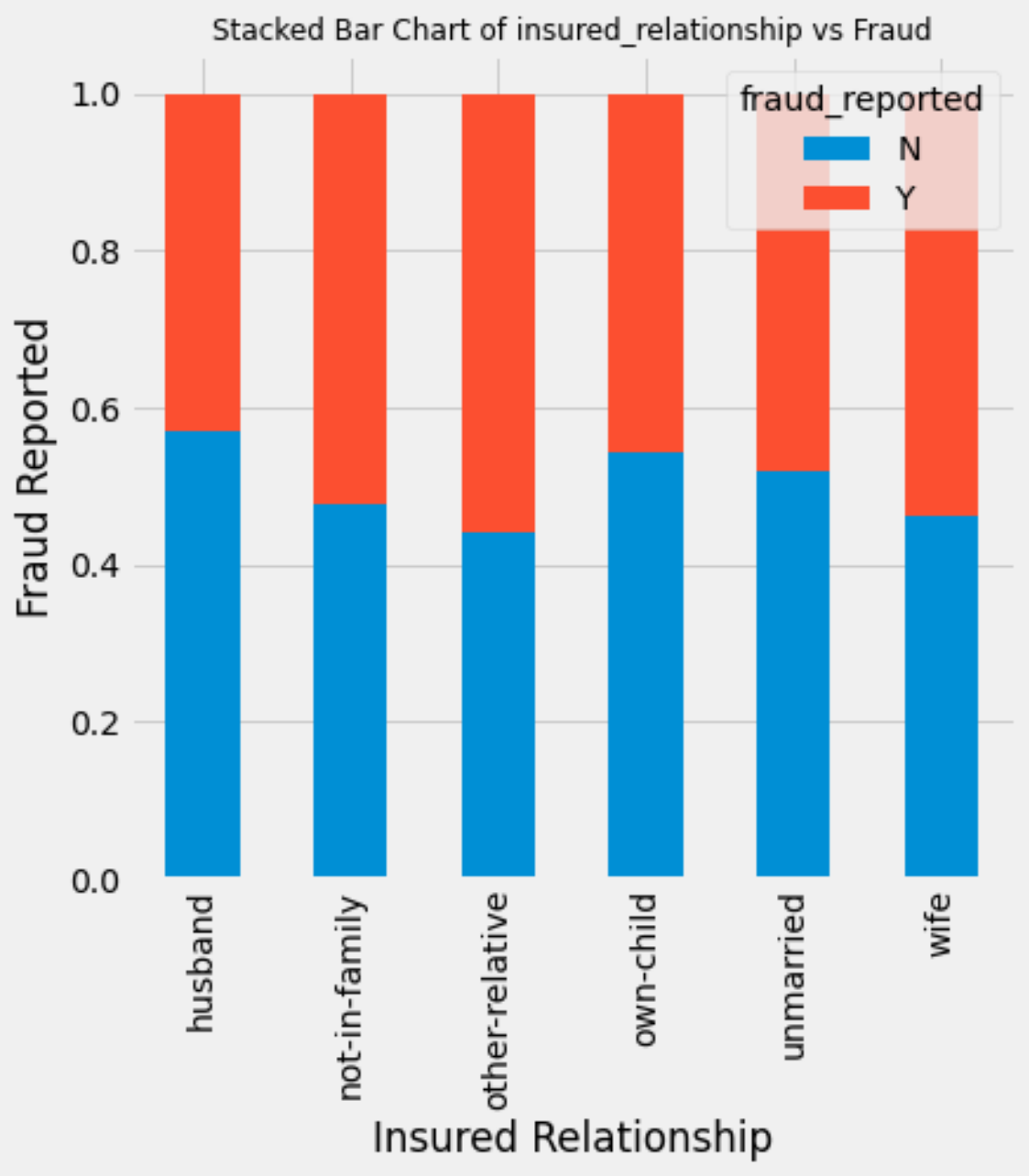
[ ]:
```
plt.rcParams['figure.figsize'] = [11, 6]
table=pd.crosstab(df.insured_sex, df.fraud_reported)
table.div(table.sum(1).astype(float), axis=0).plot(kind='bar', stacked=True)
plt.title('Stacked Bar Chart of insured_sex vs Fraud', fontsize=12)
plt.xlabel('Insured Sex')
plt.ylabel('Fraud Reported')
plt.show()
```



[ ]:
```
plt.rcParams['figure.figsize'] = [6, 6]
ax = (df['insured_relationship'].value_counts()*100.0 /len(df))\
.plot.pie(autopct='%.1f%%', labels = ['husband', 'wife', 'own-child',␣
 ↪'unmarried', 'other-relative', 'not-in-family'],
         fontsize=12)
ax.set_title('% Relationship')
plt.ylabel('Insured Relationship')
plt.show()
```

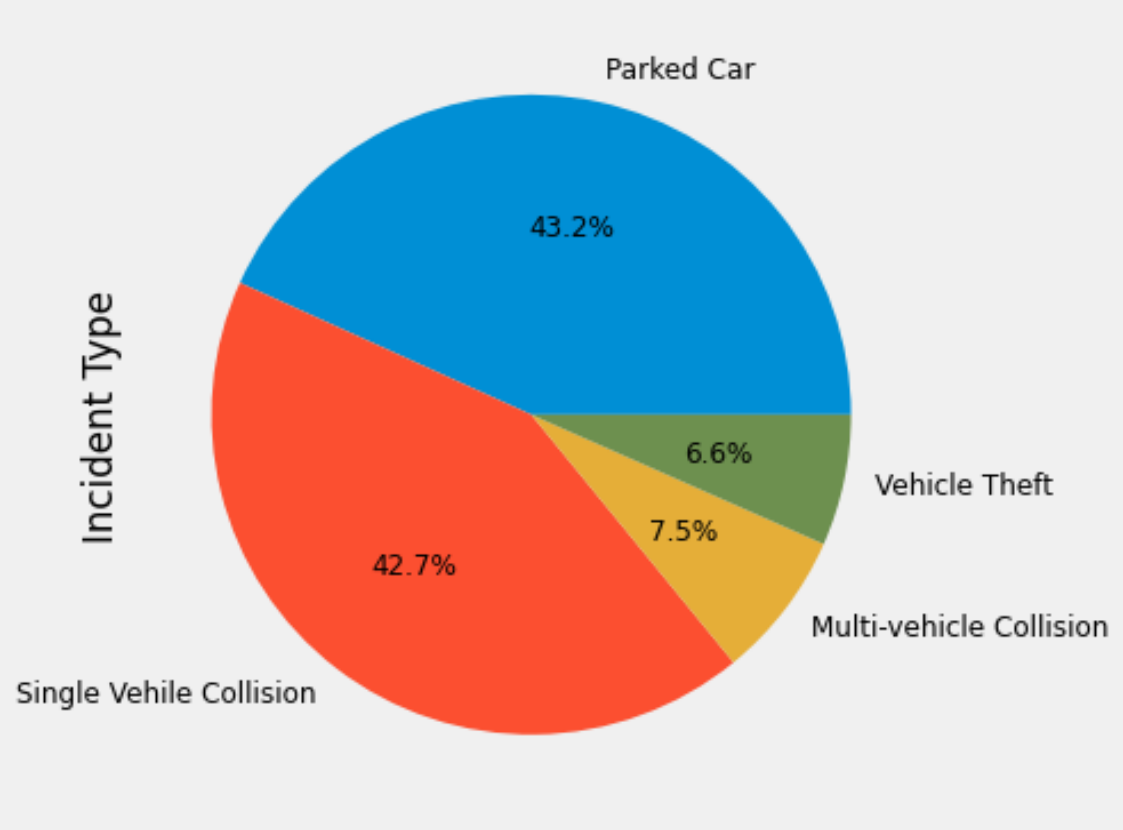


[ ]:
```
table=pd.crosstab(df.insured_relationship, df.fraud_reported)
table.div(table.sum(1).astype(float), axis=0).plot(kind='bar', stacked=True)
plt.title('Stacked Bar Chart of insured_relationship vs Fraud', fontsize=12)
plt.xlabel('Insured Relationship')
plt.ylabel('Fraud Reported')
plt.show()
```



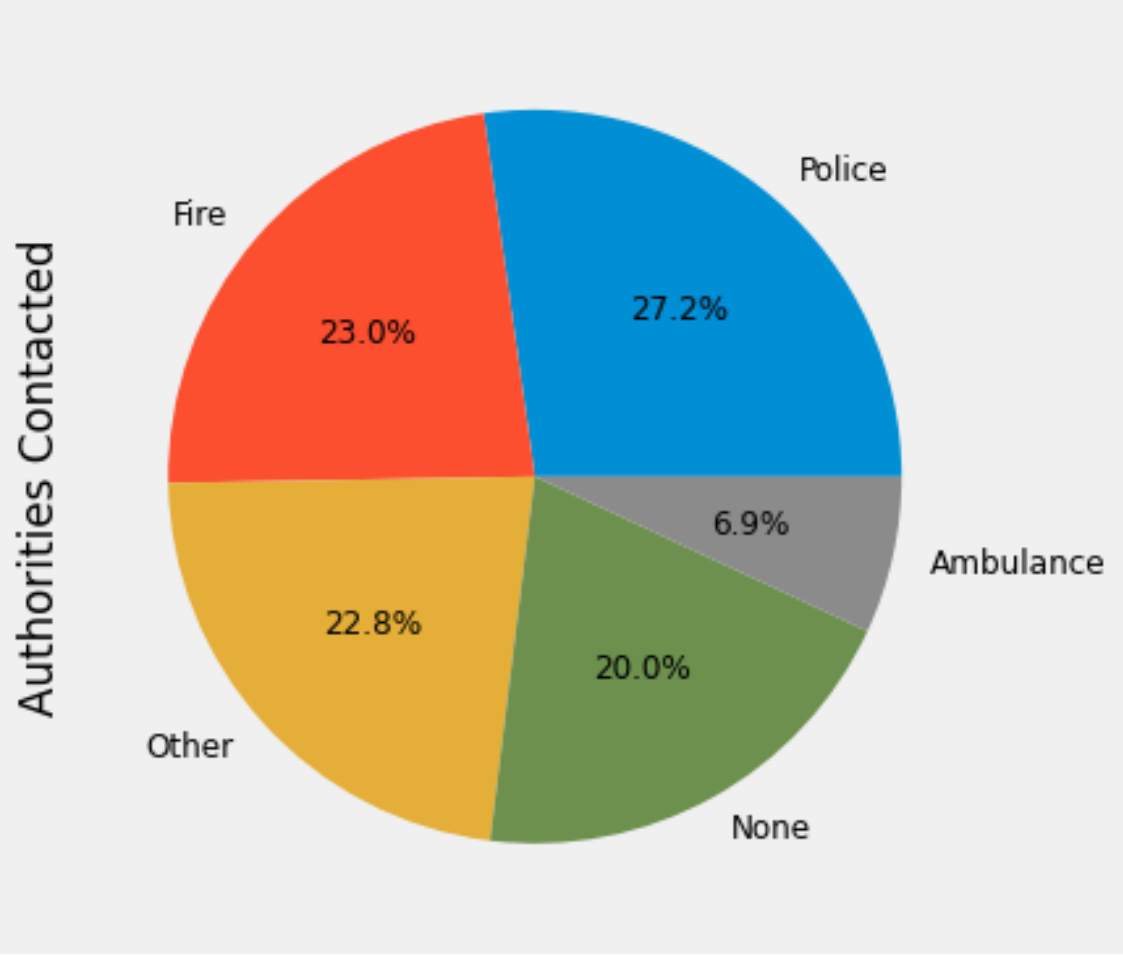
[ ]:
```
fig = plt.figure(figsize=(6,6))
ax = (df['incident_type'].value_counts()*100.0 /len(df))\
.plot.pie(autopct='%.1f%%', labels = ['Parked Car', 'Single Vehile Collision',␣
↪'Multi-vehicle Collision', 'Vehicle Theft'],
          fontsize=12)
plt.ylabel('Incident Type')
```

[ ]:
```
Text(0, 0.5, 'Incident Type')
```

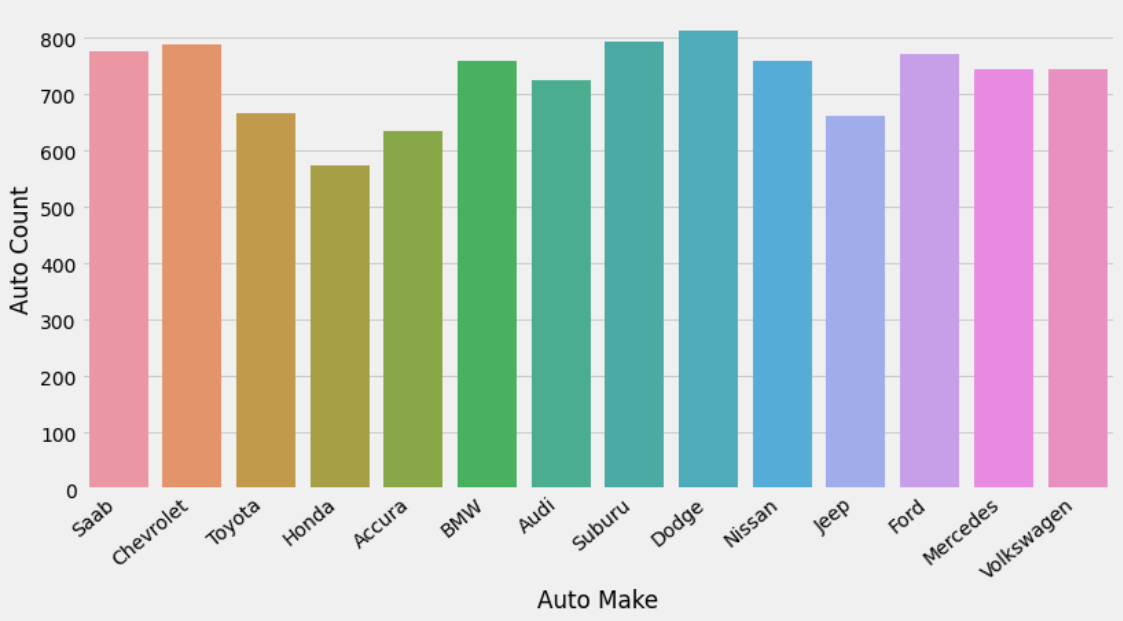


[ ]:
```
fig = plt.figure(figsize=(6,6))
ax = (df['authorities_contacted'].value_counts()*100.0 /len(df))\
.plot.pie(autopct='%.1f%%', labels = ['Police', 'Fire', 'Other', 'None',␣
↪'Ambulance'],
          fontsize=12)
plt.ylabel('Authorities Contacted')
```

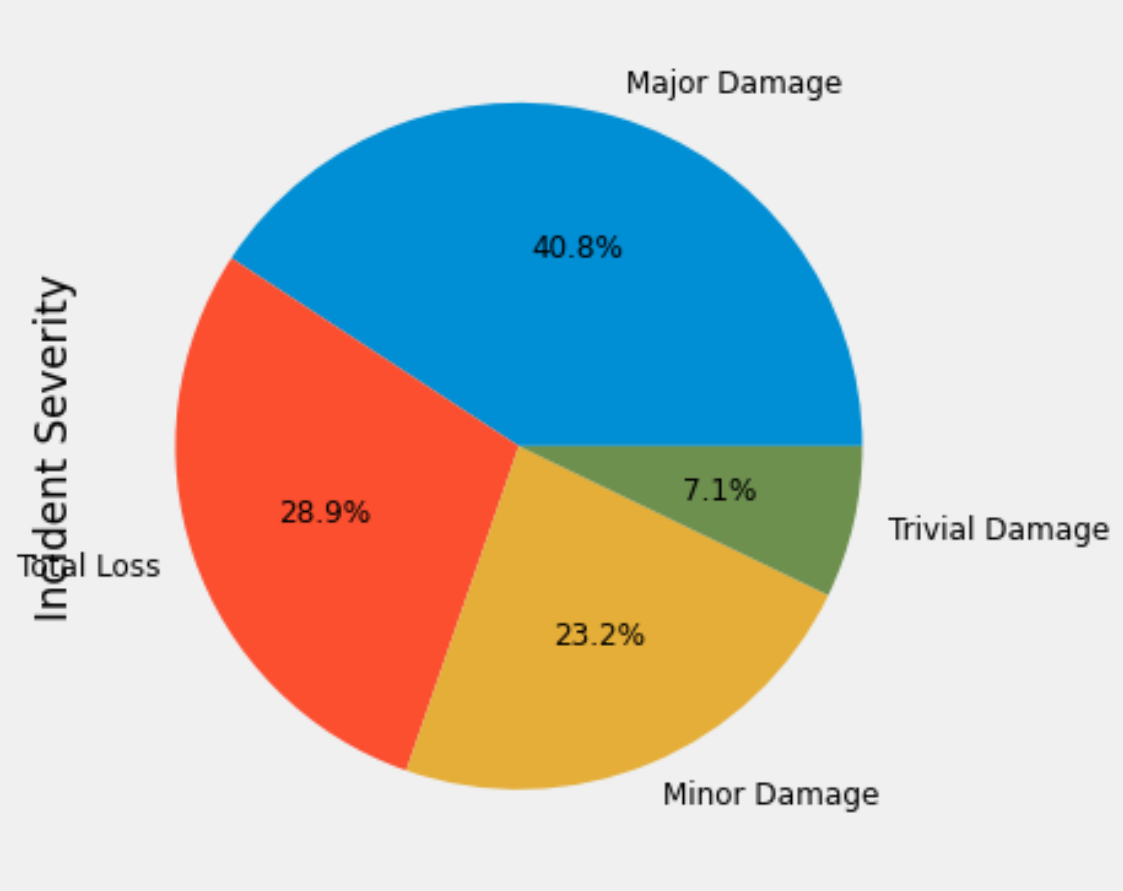
[ ]:
```
Text(0, 0.5, 'Authorities Contacted')
```



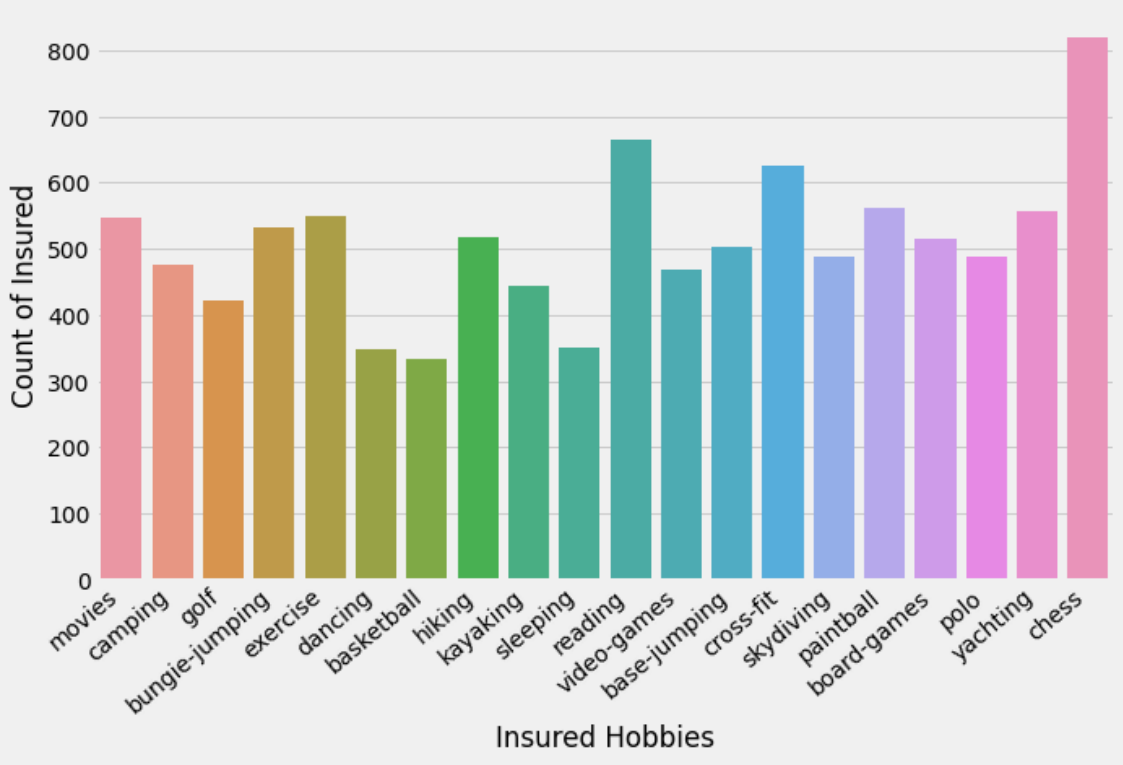
[ ]:
```
fig = plt.figure(figsize=(12,6))
ax = sns.countplot(x='auto_make', data=df)
ax.set_xticklabels(ax.get_xticklabels(), rotation=40, ha="right")
plt.xlabel('Auto Make')
plt.ylabel('Auto Count')
plt.show()
```



[ ]:
```
fig = plt.figure(figsize=(6,6))
ax = (df['incident_severity'].value_counts()*100.0 /len(df))\
.plot.pie(autopct='%.1f%%', labels = ['Major Damage', 'Total Loss', 'Minor␣
 ↪Damage', 'Trivial Damage'],
          fontsize=12)
plt.ylabel('Incident Severity');
```



[ ]:
```
fig = plt.figure(figsize=(10,6))
ax = sns.countplot(x='insured_hobbies', data=df)
ax.set_xticklabels(ax.get_xticklabels(), rotation=40, ha="right")
plt.xlabel('Insured Hobbies')
plt.ylabel('Count of Insured')
plt.show()
```

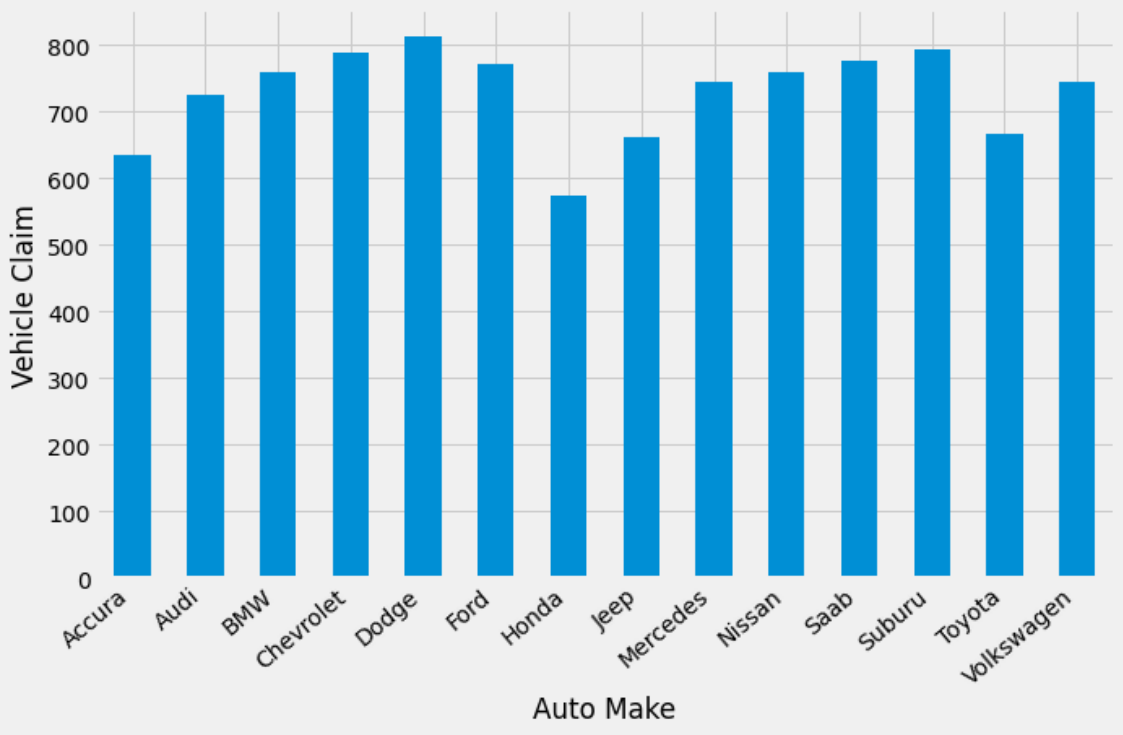


[ ]:
```
df["insured_occupation"].value_counts()
```

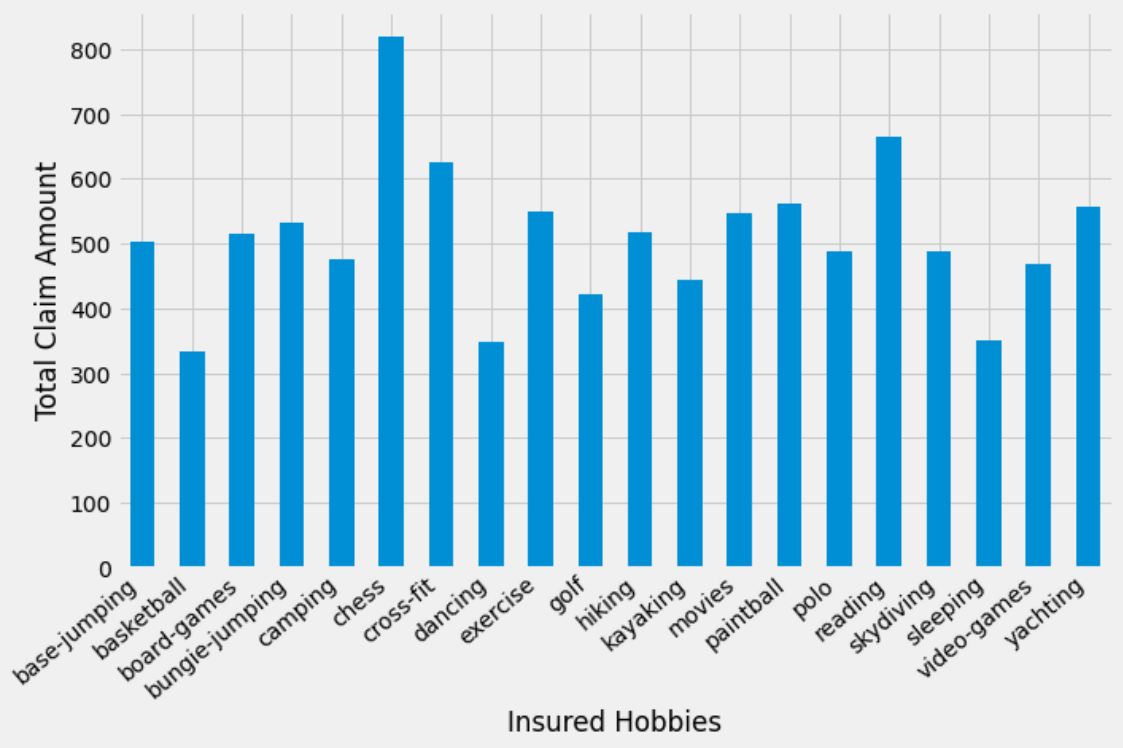
[ ]:
```
machine-op-inspct    894
exec-managerial      888
tech-support         851
prof-specialty       845
craft-repair         783
sales                780
armed-forces         731
transport-moving     730
priv-house-serv      654
protective-serv      646
other-service        641
adm-clerical         618
farming-fishing      597
handlers-cleaners    553
Name: insured_occupation, dtype: int64
```

[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(10,6))
ax= df.groupby('auto_make').vehicle_claim.count().plot.bar(ylim=0)
ax.set_ylabel('Vehicle Claim')
ax.set_xlabel('Auto Make')
```

```
ax.set_xticklabels(ax.get_xticklabels(), rotation=40, ha="right")
plt.show()
```



[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(10,6))
ax= df.groupby('insured_hobbies').total_claim_amount.count().plot.bar(ylim=0)
ax.set_ylabel('Total Claim Amount')
ax.set_xlabel('Insured Hobbies')
ax.set_xticklabels(ax.get_xticklabels(), rotation=40, ha="right")
plt.show()
```



3 Data Processing

Cleaning up the data and prepare it for machine learning model.

[ ]:
```
df['fraud_reported'].replace(to_replace='Y', value=1, inplace=True)
df['fraud_reported'].replace(to_replace='N',  value=0, inplace=True)

df.head()
```

[ ]:

	months_as_customer	age	policy_number	policy_bind_date	policy_state	\
0	5	37	939011	16-07-2002	IN	
1	462	58	902576	28-11-2002	IL	
2	198	51	575784	12-05-2007	OH	
3	384	47	102488	10-02-1998	OH	
4	100	27	1129102	01-03-2012	IL	

	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit	\
0	250/500	500	1145.28	0.0	
1	500/1000	1000	1156.80	0.0	
2	100/300	2000	751.02	0.0	
3	100/300	500	1137.34	1000000.0	
4	100/300	2000	1082.70	4000000.0	

```
   insured_zip insured_sex insured_education_level insured_occupation  \
0       360963      FEMALE               Associate    priv-house-serv
1       432568      FEMALE                     MD    exec-managerial
2       712296      FEMALE             High School    farming-fishing
3       402197      FEMALE             High School   transport-moving
4       577005      FEMALE                     PhD       armed-forces

  insured_hobbies insured_relationship  capital.gains  capital.loss  \
0          movies              husband          54735         88553
1         camping       other-relative           1381         50621
2            golf            own-child              0             0
3  bungie-jumping              husband              0         42211
4        exercise              husband              0             0

  incident_date            incident_type   collision_type incident_severity  \
0    06-02-2015  Single Vehicle Collision  Front Collision      Minor Damage
1    18-01-2015   Multi-vehicle Collision   Rear Collision        Total Loss
2    13-02-2015               Parked Car                ?    Trivial Damage
3    27-01-2015            Vehicle Theft                ?    Trivial Damage
4    21-02-2015            Vehicle Theft                ?      Minor Damage

  authorities_contacted incident_state incident_city   incident_location  \
0                 Other             NY     Hillsdale         6770 1st St
1                 Other             SC     Arlington        3275 Pine St
2                  None             SC     Arlington     1741 Best Ridge
3                Police             WV   Springfield    9744 Texas Drive
4                  None             OH    Northbrook  3289 Britain Drive

   incident_hour_of_the_day  number_of_vehicles_involved property_damage  \
0                        20                            1               ?
1                        11                            2               ?
2                         0                            1              NO
3                         6                            1             YES
4                         5                            1              NO

   bodily_injuries  witnesses police_report_available  total_claim_amount  \
0                2          1                     YES             96200.0
1                0          5                       ?             31200.0
2                1          3                       ?             14500.0
3                1          1                      NO              7500.0
4                2          1                     YES             16500.0

   injury_claim  property_claim  vehicle_claim  auto_make  auto_model  \
0          3000             500          58870       Saab         92x
1          3830            7370          32130       Saab          95
2             0               0           5690  Chevrolet      Malibu
```

```
3              0                0            420       Saab          95
4           5400             4300           8270     Toyota  Highlander

   auto_year  fraud_reported incident_state_count
0       1997               0                 262
1       2006               0                 248
2       1996               0                 248
3       1990               0                 217
4       1998               0                  23
```

[ ]:
```
df[['insured_zip']] = df[['insured_zip']].astype(object)
df.describe()
```

[ ]:
```
       months_as_customer           age  policy_number  policy_deductable  \
count        10211.000000  10211.000000   1.021100e+04       10211.000000
mean           213.467927     39.050142   5.474680e+05        1159.044168
std            133.639732     11.508964   3.034069e+05         621.773731
min              0.000000      2.000000   4.410000e+02         500.000000
25%            106.000000     31.000000   3.095050e+05         500.000000
50%            202.000000     38.000000   5.364750e+05        1000.000000
75%            303.000000     47.000000   7.717955e+05        2000.000000
max            747.000000     79.000000   1.615353e+06        2000.000000

       policy_annual_premium  umbrella_limit  capital.gains   capital.loss  \
count           10211.000000    1.021100e+04   10211.000000   10211.000000
mean             1257.794204    1.727157e+06   16459.434531   17150.482029
std               300.874661    2.407828e+06   24596.437735   25528.629014
min               179.790000    0.000000e+00       0.000000       0.000000
25%              1058.790000    0.000000e+00       0.000000       0.000000
50%              1255.210000    1.000000e+06     155.000000       0.000000
75%              1454.130000    3.000000e+06   26191.000000   27564.000000
max              2390.510000    1.000000e+07  134607.000000  142321.000000

       incident_hour_of_the_day  number_of_vehicles_involved  bodily_injuries  \
count              10211.000000                 10211.000000     10211.000000
mean                  11.229850                     2.010087         1.139262
std                    6.411803                     1.101773         0.896285
min                    0.000000                     1.000000         0.000000
25%                    6.000000                     1.000000         0.000000
50%                   12.000000                     2.000000         1.000000
75%                   16.000000                     3.000000         2.000000
max                   24.000000                     6.000000         4.000000

          witnesses  total_claim_amount  injury_claim  property_claim  \
count  10211.000000        10211.000000  10211.000000    10211.000000
mean       1.652434        56608.934482   7912.681422     8028.269513
std        1.195957        27647.190092   5456.281497     5514.767560
```

```
min        0.000000          100.000000      0.000000        0.000000
25%        1.000000        37930.000000   3615.000000     3730.000000
50%        2.000000        58200.000000   7460.000000     7600.000000
75%        3.000000        76000.000000  11800.000000    11770.000000
max        6.000000       154740.000000  30000.000000    29700.000000

       vehicle_claim     auto_year  fraud_reported
count   10211.000000  10211.000000    10211.000000
mean    40822.630497   2004.358927        0.498776
std     19666.958809      6.442418        0.500023
min        10.000000   1981.000000        0.000000
25%     27700.000000   2000.000000        0.000000
50%     42200.000000   2005.000000        0.000000
75%     54700.000000   2009.000000        1.000000
max    110800.000000   2015.000000        1.000000
```

Some variables such as 'policy_bind_date', 'incident_date', 'incident_location' and 'insured_zip' contain very high number of level. We will remove these columns for our purposes.

Let's view summary of all the column with the object data-type :

[ ]:
```
df.describe(include='all')
```

[ ]:
```
        months_as_customer           age  policy_number policy_bind_date  \
count         10211.000000  10211.000000   1.021100e+04             10211
unique                 NaN           NaN            NaN               951
top                    NaN           NaN            NaN        07-11-1997
freq                   NaN           NaN            NaN                61
mean            213.467927     39.050142   5.474680e+05               NaN
std             133.639732     11.508964   3.034069e+05               NaN
min               0.000000      2.000000   4.410000e+02               NaN
25%             106.000000     31.000000   3.095050e+05               NaN
50%             202.000000     38.000000   5.364750e+05               NaN
75%             303.000000     47.000000   7.717955e+05               NaN
max             747.000000     79.000000   1.615353e+06               NaN

       policy_state policy_csl  policy_deductable  policy_annual_premium  \
count         10211      10211       10211.000000           10211.000000
unique            3          3                NaN                    NaN
top              OH    250/500                NaN                    NaN
freq           3549       3671                NaN                    NaN
mean            NaN        NaN        1159.044168            1257.794204
std             NaN        NaN         621.773731             300.874661
min             NaN        NaN         500.000000             179.790000
25%             NaN        NaN         500.000000            1058.790000
50%             NaN        NaN        1000.000000            1255.210000
75%             NaN        NaN        2000.000000            1454.130000
max             NaN        NaN        2000.000000            2390.510000
```

	umbrella_limit	insured_zip	insured_sex	insured_education_level
count	1.021100e+04	10211.0	10211	10211
unique	NaN	10045.0	2	7
top	NaN	429118.0	FEMALE	JD
freq	NaN	2.0	5451	1687
mean	1.727157e+06	NaN	NaN	NaN
std	2.407828e+06	NaN	NaN	NaN
min	0.000000e+00	NaN	NaN	NaN
25%	0.000000e+00	NaN	NaN	NaN
50%	1.000000e+06	NaN	NaN	NaN
75%	3.000000e+06	NaN	NaN	NaN
max	1.000000e+07	NaN	NaN	NaN

	insured_occupation	insured_hobbies	insured_relationship	capital.gains
count	10211	10211	10211	10211.000000
unique	14	20	6	NaN
top	machine-op-inspct	chess	other-relative	NaN
freq	894	819	1864	NaN
mean	NaN	NaN	NaN	16459.434531
std	NaN	NaN	NaN	24596.437735
min	NaN	NaN	NaN	0.000000
25%	NaN	NaN	NaN	0.000000
50%	NaN	NaN	NaN	155.000000
75%	NaN	NaN	NaN	26191.000000
max	NaN	NaN	NaN	134607.000000

	capital.loss	incident_date	incident_type	collision_type
count	10211.000000	10211	10211	10211
unique	NaN	60	4	4
top	NaN	02-02-2015	Single Vehicle Collision	Rear Collision
freq	NaN	311	4415	3283
mean	17150.482029	NaN	NaN	NaN
std	25528.629014	NaN	NaN	NaN
min	0.000000	NaN	NaN	NaN
25%	0.000000	NaN	NaN	NaN
50%	0.000000	NaN	NaN	NaN
75%	27564.000000	NaN	NaN	NaN
max	142321.000000	NaN	NaN	NaN

	incident_severity	authorities_contacted	incident_state	incident_city
count	10211	10211	10211	10211
unique	4	5	7	7
top	Major Damage	Police	SC	Springfield
freq	4164	2781	2656	1599
mean	NaN	NaN	NaN	NaN
std	NaN	NaN	NaN	NaN

```
min                  NaN                    NaN            NaN           NaN
25%                  NaN                    NaN            NaN           NaN
50%                  NaN                    NaN            NaN           NaN
75%                  NaN                    NaN            NaN           NaN
max                  NaN                    NaN            NaN           NaN

       incident_location  incident_hour_of_the_day  \
count              10211              10211.000000
unique              1000                       NaN
top     8954 Apache Lane                       NaN
freq                  34                       NaN
mean                 NaN                 11.229850
std                  NaN                  6.411803
min                  NaN                  0.000000
25%                  NaN                  6.000000
50%                  NaN                 12.000000
75%                  NaN                 16.000000
max                  NaN                 24.000000

       number_of_vehicles_involved property_damage  bodily_injuries  \
count                 10211.000000           10211     10211.000000
unique                         NaN               3              NaN
top                            NaN               ?              NaN
freq                           NaN            3869              NaN
mean                      2.010087             NaN         1.139262
std                       1.101773             NaN         0.896285
min                       1.000000             NaN         0.000000
25%                       1.000000             NaN         0.000000
50%                       2.000000             NaN         1.000000
75%                       3.000000             NaN         2.000000
max                       6.000000             NaN         4.000000

          witnesses police_report_available  total_claim_amount  \
count   10211.000000                  10211        10211.000000
unique           NaN                      3                 NaN
top              NaN                      ?                 NaN
freq             NaN                   3577                 NaN
mean        1.652434                    NaN        56608.934482
std         1.195957                    NaN        27647.190092
min         0.000000                    NaN          100.000000
25%         1.000000                    NaN        37930.000000
50%         2.000000                    NaN        58200.000000
75%         3.000000                    NaN        76000.000000
max         6.000000                    NaN       154740.000000

       injury_claim  property_claim  vehicle_claim auto_make auto_model  \
count   10211.000000    10211.000000   10211.000000     10211      10211
```

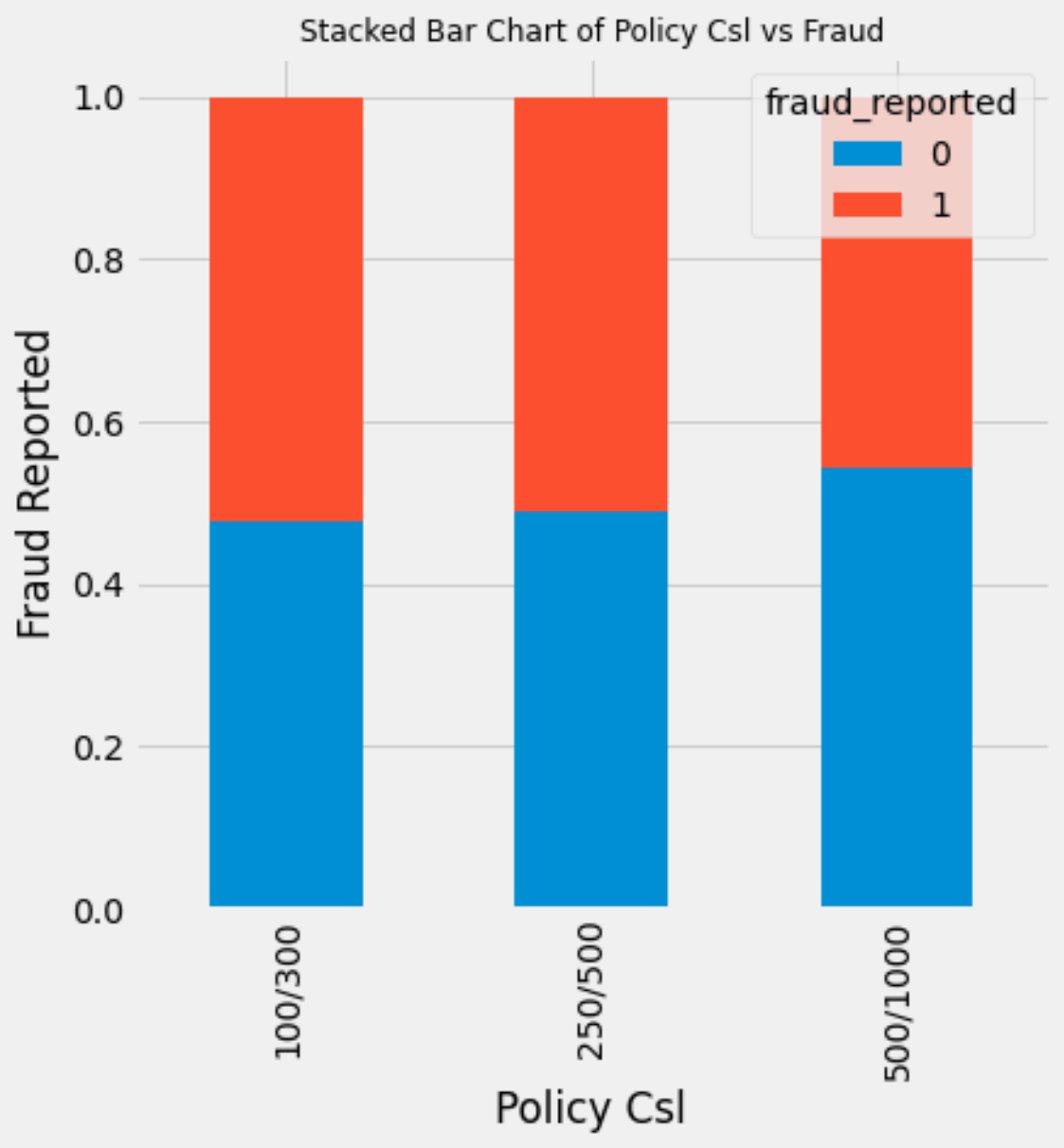
unique	NaN	NaN	NaN	14	39
top	NaN	NaN	NaN	Dodge	RAM
freq	NaN	NaN	NaN	813	485
mean	7912.681422	8028.269513	40822.630497	NaN	NaN
std	5456.281497	5514.767560	19666.958809	NaN	NaN
min	0.000000	0.000000	10.000000	NaN	NaN
25%	3615.000000	3730.000000	27700.000000	NaN	NaN
50%	7460.000000	7600.000000	42200.000000	NaN	NaN
75%	11800.000000	11770.000000	54700.000000	NaN	NaN
max	30000.000000	29700.000000	110800.000000	NaN	NaN

	auto_year	fraud_reported	incident_state_count
count	10211.000000	10211.000000	10211.0
unique	NaN	NaN	6.0
top	NaN	NaN	248.0
freq	NaN	NaN	2656.0
mean	2004.358927	0.498776	NaN
std	6.442418	0.500023	NaN
min	1981.000000	0.000000	NaN
25%	2000.000000	0.000000	NaN
50%	2005.000000	0.000000	NaN
75%	2009.000000	1.000000	NaN
max	2015.000000	1.000000	NaN

Some values in the table are shown here as "NaN". We will see how to deal with these missing values.

[ ]:
```
plt.style.use('fivethirtyeight')
plt.figure(figsize=(14,6))
table=pd.crosstab(df.policy_csl, df.fraud_reported)
table.div(table.sum(1).astype(float), axis=0).plot(kind='bar', stacked=True)
plt.title('Stacked Bar Chart of Policy Csl vs Fraud', fontsize=12)
plt.xlabel('Policy Csl')
plt.ylabel('Fraud Reported')
plt.show();
```

```
<Figure size 1008x432 with 0 Axes>
```



policy_csl looks like an unavidable predictor.

[ ]:
```
df['csl_per_person'] = df.policy_csl.str.split('/', expand=True)[0]
df['csl_per_accident'] = df.policy_csl.str.split('/', expand=True)[1]
df['csl_per_person'].head()
```

[ ]:
```
0    250
1    500
2    100
3    100
4    100
```

```
Name: csl_per_person, dtype: object
```

[ ]:
```
df['csl_per_accident'].head()
```

[ ]:
```
0     500
1    1000
2     300
3     300
4     300
Name: csl_per_accident, dtype: object
```

[ ]:
```
df.auto_year.value_counts()  # check the spread of years to decide on further␣
↪action.
```

[ ]:
```
2008    905
2010    667
2009    587
2005    515
2007    484
2004    472
2003    465
2006    460
2002    455
2011    437
1999    435
2001    426
2012    423
2000    418
2013    379
2014    361
1998    352
1997    347
1996    279
1995    276
2015    239
1994    204
1993    181
1992    156
1991    105
1990     72
1989     40
1988     26
1987     17
1986     15
1984      6
1985      3
1983      2
```

```
1981      1
1982      1
Name: auto_year, dtype: int64
```

auto_year has 21 levels, and the number of records for each of the levels are quite significant considering datasize is not so large. We will do some feature engineering using this variable considering, the year of manufacturing of automobile indicates the age of the vehicle and may contain valuable information for insurance premium or fraud is concerned.

[ ]:
```
df['vehicle_age'] = 2018 - df['auto_year'] # Deriving the age of the vehicle␣
↪based on the year value
df['vehicle_age'].head(10)
```

[ ]:
```
0    21
1    12
2    22
3    28
4    20
5    23
6    13
7     9
8    13
9    18
Name: vehicle_age, dtype: int64
```

[ ]:
```
bins = [-1, 3, 6, 9, 12, 17, 20, 24]  # Factorize according to the time period␣
↪of the day.
names = ["past_midnight", "early_morning", "morning", 'fore-noon', 'afternoon',␣
↪'evening', 'night']
df['incident_period_of_day'] = pd.cut(df.incident_hour_of_the_day, bins,␣
↪labels=names).astype(object)
df[['incident_hour_of_the_day', 'incident_period_of_day']].head(20)
```

[ ]:

	incident_hour_of_the_day	incident_period_of_day
0	20	evening
1	11	fore-noon
2	0	past_midnight
3	6	early_morning
4	5	early_morning
5	6	early_morning
6	11	fore-noon
7	3	past_midnight
8	22	night
9	16	afternoon
10	15	afternoon
11	21	night
12	14	afternoon

```
13                      15            afternoon
14                       6        early_morning
15                       0        past_midnight
16                       4        early_morning
17                       8              morning
18                      13            afternoon
19                       9              morning
```

[ ]:
```
# Check on categorical variables:
df.select_dtypes(include=['object']).columns  # checking categorcial columns
```

[ ]:
```
Index(['policy_bind_date', 'policy_state', 'policy_csl', 'insured_zip',
       'insured_sex', 'insured_education_level', 'insured_occupation',
       'insured_hobbies', 'insured_relationship', 'incident_date',
       'incident_type', 'collision_type', 'incident_severity',
       'authorities_contacted', 'incident_state', 'incident_city',
       'incident_location', 'property_damage', 'police_report_available',
       'auto_make', 'auto_model', 'incident_state_count', 'csl_per_person',
       'csl_per_accident', 'incident_period_of_day'],
      dtype='object')
```

[ ]:
```
# dropping unimportant columns

df = df.drop(columns = [
    'policy_number',
    'policy_csl',
    'insured_zip',
    'policy_bind_date',
    'incident_date',
    'incident_location',
    'auto_year',
    'incident_hour_of_the_day'])

df.head(2)
```

[ ]:
```
   months_as_customer  age policy_state  policy_deductable  \
0                   5   37           IN                500
1                 462   58           IL               1000

   policy_annual_premium  umbrella_limit insured_sex insured_education_level  \
0                1145.28             0.0      FEMALE                Associate
1                1156.80             0.0      FEMALE                       MD

  insured_occupation insured_hobbies insured_relationship  capital.gains  \
0    priv-house-serv          movies              husband          54735
1    exec-managerial         camping       other-relative           1381
```

```
   capital.loss             incident_type   collision_type incident_severity  \
0         88553  Single Vehicle Collision  Front Collision      Minor Damage
1         50621   Multi-vehicle Collision   Rear Collision        Total Loss

  authorities_contacted incident_state incident_city  \
0                 Other             NY     Hillsdale
1                 Other             SC     Arlington

   number_of_vehicles_involved property_damage  bodily_injuries  witnesses  \
0                            1               ?                2          1
1                            2               ?                0          5

  police_report_available  total_claim_amount  injury_claim  property_claim  \
0                     YES             96200.0          3000             500
1                       ?             31200.0          3830            7370

   vehicle_claim auto_make auto_model  fraud_reported incident_state_count  \
0          58870      Saab        92x               0                  262
1          32130      Saab         95               0                  248

  csl_per_person csl_per_accident  vehicle_age incident_period_of_day
0            250              500           21                evening
1            500             1000           12              fore-noon
```

[ ]:
```
# identify variables with '?' values
unknowns = {}
for i in list(df.columns):
    if (df[i]).dtype == object:
        j = np.sum(df[i] == "?")
        unknowns[i] = j
unknowns = pd.DataFrame.from_dict(unknowns, orient = 'index')
print(unknowns)
```

```
                            0
policy_state                0
insured_sex                 0
insured_education_level     0
insured_occupation          0
insured_hobbies             0
insured_relationship        0
incident_type               0
collision_type           1440
incident_severity           0
authorities_contacted       0
incident_state              0
incident_city               0
property_damage          3869
```

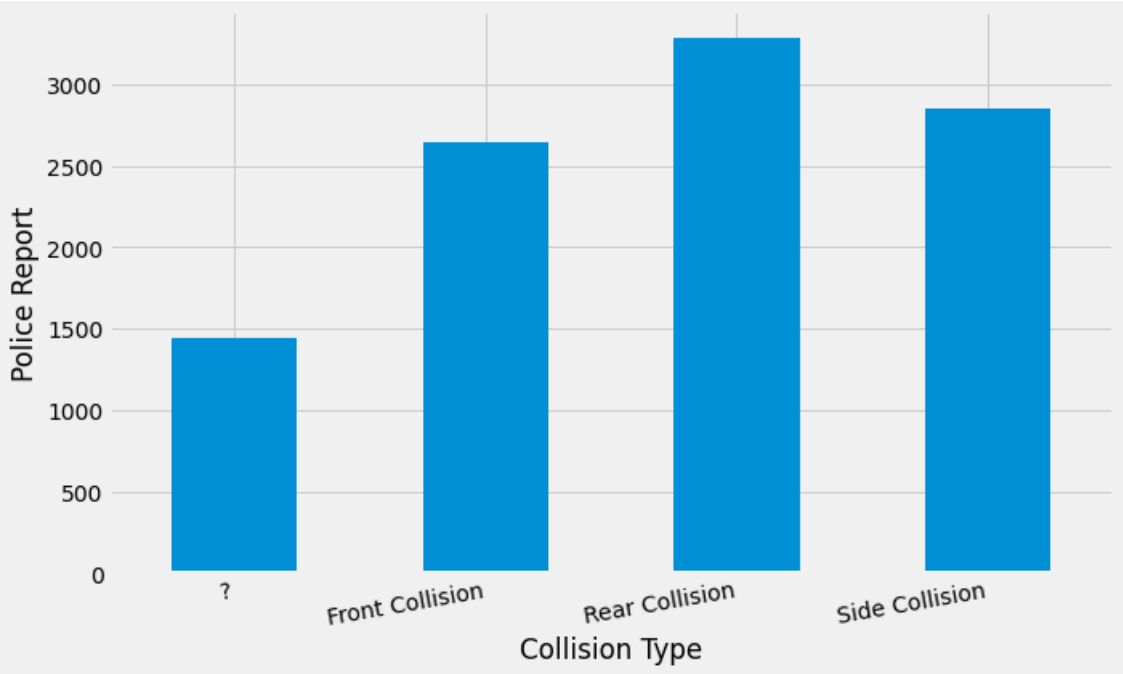
```
police_report_available  3577
auto_make                   0
auto_model                  0
incident_state_count        0
csl_per_person              0
csl_per_accident            0
incident_period_of_day      0
```

collision_type, property_damage, police_report_available contain many missing values. So, first isolate these variables, inspect these individually for spread of category values.

[ ]:
```
df.collision_type.value_counts()
```

[ ]:
```
Rear Collision     3283
Side Collision     2849
Front Collision    2639
?                  1440
Name: collision_type, dtype: int64
```

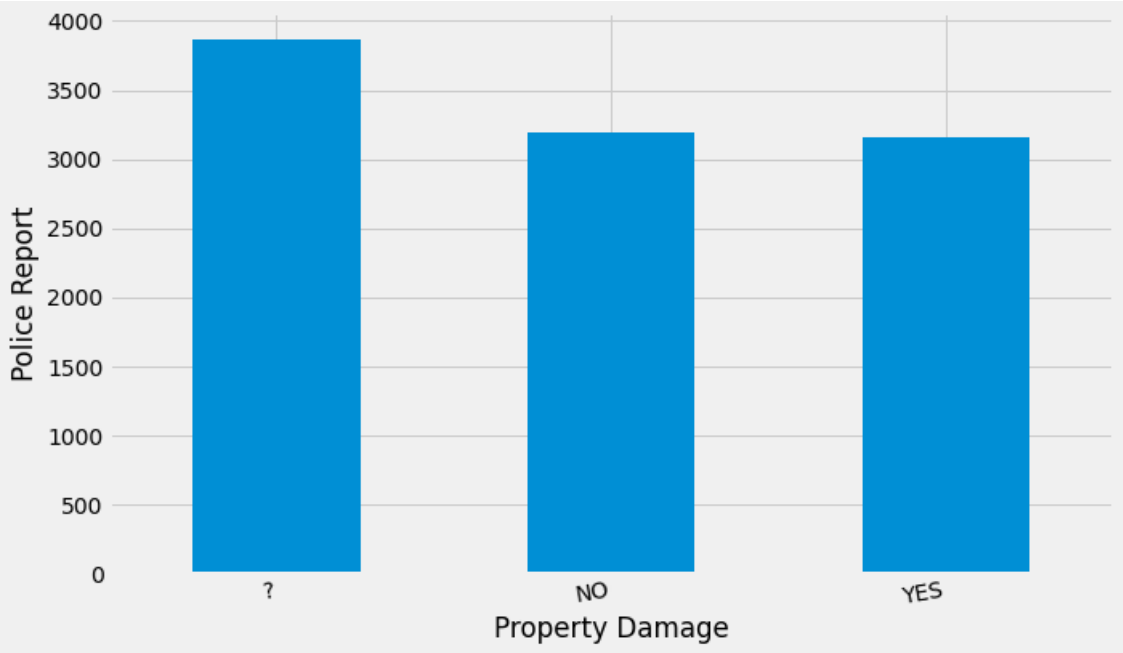
[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(10,6))
ax= df.groupby('collision_type').police_report_available.count().plot.
 ↪bar(ylim=0)
ax.set_ylabel('Police Report')
ax.set_xlabel('Collision Type')
ax.set_xticklabels(ax.get_xticklabels(), rotation=10, ha="right")
plt.show()
```



[ ]:
```
df.property_damage.value_counts()
```

[ ]:
```
?      3869
NO     3188
YES    3154
Name: property_damage, dtype: int64
```

[ ]:
```
plt.style.use('fivethirtyeight')
fig = plt.figure(figsize=(10,6))
ax= df.groupby('property_damage').police_report_available.count().plot.
 ↪bar(ylim=0)
ax.set_ylabel('Police Report')
ax.set_xlabel('Property Damage')
ax.set_xticklabels(ax.get_xticklabels(), rotation=10, ha="right")
plt.show()
```



[ ]:
```
df.police_report_available.value_counts()
```

[ ]:
```
?      3577
NO     3533
YES    3101
Name: police_report_available, dtype: int64
```

[ ]:
```
df.columns
```

[ ]:
```
Index(['months_as_customer', 'age', 'policy_state', 'policy_deductable',
       'policy_annual_premium', 'umbrella_limit', 'insured_sex',
       'insured_education_level', 'insured_occupation', 'insured_hobbies',
       'insured_relationship', 'capital.gains', 'capital.loss',
       'incident_type', 'collision_type', 'incident_severity',
       'authorities_contacted', 'incident_state', 'incident_city',
       'number_of_vehicles_involved', 'property_damage', 'bodily_injuries',
       'witnesses', 'police_report_available', 'total_claim_amount',
       'injury_claim', 'property_claim', 'vehicle_claim', 'auto_make',
       'auto_model', 'fraud_reported', 'incident_state_count',
       'csl_per_person', 'csl_per_accident', 'vehicle_age',
       'incident_period_of_day'],
      dtype='object')
```

[ ]:
```
df._get_numeric_data().head()  # Checking numeric columns
```

[ ]:

	months_as_customer	age	policy_deductable	policy_annual_premium	\
0	5	37	500	1145.28	
1	462	58	1000	1156.80	
2	198	51	2000	751.02	
3	384	47	500	1137.34	
4	100	27	2000	1082.70	

	umbrella_limit	capital.gains	capital.loss	number_of_vehicles_involved	\
0	0.0	54735	88553	1	
1	0.0	1381	50621	2	
2	0.0	0	0	1	
3	1000000.0	0	42211	1	
4	4000000.0	0	0	1	

	bodily_injuries	witnesses	total_claim_amount	injury_claim	\
0	2	1	96200.0	3000	
1	0	5	31200.0	3830	
2	1	3	14500.0	0	
3	1	1	7500.0	0	
4	2	1	16500.0	5400	

	property_claim	vehicle_claim	fraud_reported	vehicle_age
0	500	58870	0	21
1	7370	32130	0	12
2	0	5690	0	22
3	0	420	0	28
4	4300	8270	0	20

[ ]:
```
df._get_numeric_data().columns
```

```
[ ]: Index(['months_as_customer', 'age', 'policy_deductable',
             'policy_annual_premium', 'umbrella_limit', 'capital.gains',
             'capital.loss', 'number_of_vehicles_involved', 'bodily_injuries',
             'witnesses', 'total_claim_amount', 'injury_claim', 'property_claim',
             'vehicle_claim', 'fraud_reported', 'vehicle_age'],
            dtype='object')
```

```
[ ]: df.select_dtypes(include=['object']).columns  # checking categorcial columns
```

```
[ ]: Index(['policy_state', 'insured_sex', 'insured_education_level',
             'insured_occupation', 'insured_hobbies', 'insured_relationship',
             'incident_type', 'collision_type', 'incident_severity',
             'authorities_contacted', 'incident_state', 'incident_city',
             'property_damage', 'police_report_available', 'auto_make', 'auto_model',
             'incident_state_count', 'csl_per_person', 'csl_per_accident',
             'incident_period_of_day'],
            dtype='object')
```

Applying one-hot encoding to convert all categorical variables except out target variables 'collision_type', 'property_damage', 'police_report_available', 'fraud_reported'

```
[ ]: dummies = pd.get_dummies(df[[
         'policy_state',
         'insured_sex',
         'insured_education_level',
         'insured_occupation',
         'insured_hobbies',
         'insured_relationship',
         'incident_type',
         'incident_severity',
         'authorities_contacted',
         'incident_state',
         'incident_city',
         'auto_make',
         'auto_model',
         'csl_per_person',
         'csl_per_accident',
         'incident_period_of_day']])

     dummies = dummies.join(df[[
         'collision_type',
         'property_damage',
         'police_report_available',
         "fraud_reported"]])

     dummies.head()
```

[ ]:

	policy_state_IL	policy_state_IN	policy_state_OH	insured_sex_FEMALE	\
0	0	1	0	1	
1	1	0	0	1	
2	0	0	1	1	
3	0	0	1	1	
4	1	0	0	1	

	insured_sex_MALE	insured_education_level_Associate	\
0	0	1	
1	0	0	
2	0	0	
3	0	0	
4	0	0	

	insured_education_level_College	insured_education_level_High School	\
0	0	0	
1	0	0	
2	0	1	
3	0	1	
4	0	0	

	insured_education_level_JD	insured_education_level_MD	\
0	0	0	
1	0	1	
2	0	0	
3	0	0	
4	0	0	

	insured_education_level_Masters	insured_education_level_PhD	\
0	0	0	
1	0	0	
2	0	0	
3	0	0	
4	0	1	

	insured_occupation_adm-clerical	insured_occupation_armed-forces	\
0	0	0	
1	0	0	
2	0	0	
3	0	0	
4	0	1	

	insured_occupation_craft-repair	insured_occupation_exec-managerial	\
0	0	0	
1	0	1	
2	0	0	
3	0	0	

```
4                                 0                                     0

   insured_occupation_farming-fishing  insured_occupation_handlers-cleaners  \
0                                   0                                     0
1                                   0                                     0
2                                   1                                     0
3                                   0                                     0
4                                   0                                     0

   insured_occupation_machine-op-inspct  insured_occupation_other-service  \
0                                     0                                   0
1                                     0                                   0
2                                     0                                   0
3                                     0                                   0
4                                     0                                   0

   insured_occupation_priv-house-serv  insured_occupation_prof-specialty  \
0                                   1                                   0
1                                   0                                   0
2                                   0                                   0
3                                   0                                   0
4                                   0                                   0

   insured_occupation_protective-serv  insured_occupation_sales  \
0                                   0                         0
1                                   0                         0
2                                   0                         0
3                                   0                         0
4                                   0                         0

   insured_occupation_tech-support  insured_occupation_transport-moving  \
0                                0                                     0
1                                0                                     0
2                                0                                     0
3                                0                                     1
4                                0                                     0

   insured_hobbies_base-jumping  insured_hobbies_basketball  \
0                             0                           0
1                             0                           0
2                             0                           0
3                             0                           0
4                             0                           0

   insured_hobbies_board-games  insured_hobbies_bungie-jumping  \
0                            0                               0
1                            0                               0
```

```
2                            0                                0
3                            0                                1
4                            0                                0

   insured_hobbies_camping  insured_hobbies_chess  insured_hobbies_cross-fit  \
0                        0                      0                          0
1                        1                      0                          0
2                        0                      0                          0
3                        0                      0                          0
4                        0                      0                          0

   insured_hobbies_dancing  insured_hobbies_exercise  insured_hobbies_golf  \
0                        0                         0                     0
1                        0                         0                     0
2                        0                         0                     1
3                        0                         0                     0
4                        0                         1                     0

   insured_hobbies_hiking  insured_hobbies_kayaking  insured_hobbies_movies  \
0                       0                         0                       1
1                       0                         0                       0
2                       0                         0                       0
3                       0                         0                       0
4                       0                         0                       0

   insured_hobbies_paintball  insured_hobbies_polo  insured_hobbies_reading  \
0                          0                     0                        0
1                          0                     0                        0
2                          0                     0                        0
3                          0                     0                        0
4                          0                     0                        0

   insured_hobbies_skydiving  insured_hobbies_sleeping  \
0                          0                         0
1                          0                         0
2                          0                         0
3                          0                         0
4                          0                         0

   insured_hobbies_video-games  insured_hobbies_yachting  \
0                            0                         0
1                            0                         0
2                            0                         0
3                            0                         0
4                            0                         0

   insured_relationship_husband  insured_relationship_not-in-family  \
```

```
0                              1                                   0
1                              0                                   0
2                              0                                   0
3                              1                                   0
4                              1                                   0

   insured_relationship_other-relative  insured_relationship_own-child  \
0                                    0                               0
1                                    1                               0
2                                    0                               1
3                                    0                               0
4                                    0                               0

   insured_relationship_unmarried  insured_relationship_wife  \
0                               0                          0
1                               0                          0
2                               0                          0
3                               0                          0
4                               0                          0

   incident_type_Multi-vehicle Collision  incident_type_Parked Car  \
0                                      0                         0
1                                      1                         0
2                                      0                         1
3                                      0                         0
4                                      0                         0

   incident_type_Single Vehicle Collision  incident_type_Vehicle Theft  \
0                                       1                            0
1                                       0                            0
2                                       0                            0
3                                       0                            1
4                                       0                            1

   incident_severity_Major Damage  incident_severity_Minor Damage  \
0                               0                               1
1                               0                               0
2                               0                               0
3                               0                               0
4                               0                               1

   incident_severity_Total Loss  incident_severity_Trivial Damage  \
0                             0                                  0
1                             1                                  0
2                             0                                  1
3                             0                                  1
4                             0                                  0
```

```
   authorities_contacted_Ambulance  authorities_contacted_Fire  \
0                                0                           0
1                                0                           0
2                                0                           0
3                                0                           0
4                                0                           0

   authorities_contacted_None  authorities_contacted_Other  \
0                           0                            1
1                           0                            1
2                           1                            0
3                           0                            0
4                           1                            0

   authorities_contacted_Police  incident_state_NC  incident_state_NY  \
0                             0                 0                  1
1                             0                 0                  0
2                             0                 0                  0
3                             1                 0                  0
4                             0                 0                  0

   incident_state_OH  incident_state_PA  incident_state_SC  incident_state_VA  \
0                  0                  0                  0                  0
1                  0                  0                  1                  0
2                  0                  0                  1                  0
3                  0                  0                  0                  0
4                  1                  0                  0                  0

   incident_state_WV  incident_city_Arlington  incident_city_Columbus  \
0                  0                        0                       0
1                  0                        1                       0
2                  0                        1                       0
3                  1                        0                       0
4                  0                        0                       0

   incident_city_Hillsdale  incident_city_Northbend  incident_city_Northbrook  \
0                        1                        0                         0
1                        0                        0                         0
2                        0                        0                         0
3                        0                        0                         0
4                        0                        0                         1

   incident_city_Riverwood  incident_city_Springfield  auto_make_Accura  \
0                        0                          0                 0
1                        0                          0                 0
2                        0                          0                 0
```

```
3                       0                         1                 0
4                       0                         0                 0

   auto_make_Audi  auto_make_BMW  auto_make_Chevrolet  auto_make_Dodge  \
0               0              0                    0                0
1               0              0                    0                0
2               0              0                    1                0
3               0              0                    0                0
4               0              0                    0                0

   auto_make_Ford  auto_make_Honda  auto_make_Jeep  auto_make_Mercedes  \
0               0                0               0                   0
1               0                0               0                   0
2               0                0               0                   0
3               0                0               0                   0
4               0                0               0                   0

   auto_make_Nissan  auto_make_Saab  auto_make_Suburu  auto_make_Toyota  \
0                 0               1                 0                 0
1                 0               1                 0                 0
2                 0               0                 0                 0
3                 0               1                 0                 0
4                 0               0                 0                 1

   auto_make_Volkswagen  auto_model_3 Series  auto_model_92x  auto_model_93  \
0                     0                    0               1              0
1                     0                    0               0              0
2                     0                    0               0              0
3                     0                    0               0              0
4                     0                    0               0              0

   auto_model_95  auto_model_A3  auto_model_A5  auto_model_Accord  \
0              0              0              0                  0
1              1              0              0                  0
2              0              0              0                  0
3              1              0              0                  0
4              0              0              0                  0

   auto_model_C300  auto_model_CRV  auto_model_Camry  auto_model_Civic  \
0                0               0                 0                 0
1                0               0                 0                 0
2                0               0                 0                 0
3                0               0                 0                 0
4                0               0                 0                 0

   auto_model_Corolla  auto_model_E400  auto_model_Escape  auto_model_F150  \
0                   0                0                  0                0
```

```
1                    0                0                    0                0
2                    0                0                    0                0
3                    0                0                    0                0
4                    0                0                    0                0

   auto_model_Forrestor  auto_model_Fusion  auto_model_Grand Cherokee  \
0                     0                  0                          0
1                     0                  0                          0
2                     0                  0                          0
3                     0                  0                          0
4                     0                  0                          0

   auto_model_Highlander  auto_model_Impreza  auto_model_Jetta  \
0                      0                   0                 0
1                      0                   0                 0
2                      0                   0                 0
3                      0                   0                 0
4                      1                   0                 0

   auto_model_Legacy  auto_model_M5  auto_model_MDX  auto_model_ML350  \
0                  0              0               0                 0
1                  0              0               0                 0
2                  0              0               0                 0
3                  0              0               0                 0
4                  0              0               0                 0

   auto_model_Malibu  auto_model_Maxima  auto_model_Neon  auto_model_Passat  \
0                  0                  0                0                  0
1                  0                  0                0                  0
2                  1                  0                0                  0
3                  0                  0                0                  0
4                  0                  0                0                  0

   auto_model_Pathfinder  auto_model_RAM  auto_model_RSX  \
0                      0               0               0
1                      0               0               0
2                      0               0               0
3                      0               0               0
4                      0               0               0

   auto_model_Silverado  auto_model_TL  auto_model_Tahoe  auto_model_Ultima  \
0                     0              0                 0                  0
1                     0              0                 0                  0
2                     0              0                 0                  0
3                     0              0                 0                  0
4                     0              0                 0                  0
```

```
   auto_model_Wrangler  auto_model_X5  auto_model_X6  csl_per_person_100  \
0                    0              0              0                   0
1                    0              0              0                   0
2                    0              0              0                   1
3                    0              0              0                   1
4                    0              0              0                   1

   csl_per_person_250  csl_per_person_500  csl_per_accident_1000  \
0                   1                   0                      0
1                   0                   1                      1
2                   0                   0                      0
3                   0                   0                      0
4                   0                   0                      0

   csl_per_accident_300  csl_per_accident_500  \
0                     0                     1
1                     0                     0
2                     1                     0
3                     1                     0
4                     1                     0

   incident_period_of_day_afternoon  incident_period_of_day_early_morning  \
0                                 0                                     0
1                                 0                                     0
2                                 0                                     0
3                                 0                                     1
4                                 0                                     1

   incident_period_of_day_evening  incident_period_of_day_fore-noon  \
0                               1                                 0
1                               0                                 1
2                               0                                 0
3                               0                                 0
4                               0                                 0

   incident_period_of_day_morning  incident_period_of_day_night  \
0                               0                             0
1                               0                             0
2                               0                             0
3                               0                             0
4                               0                             0

   incident_period_of_day_past_midnight   collision_type property_damage  \
0                                     0  Front Collision               ?
1                                     0   Rear Collision               ?
2                                     1                ?              NO
3                                     0                ?             YES
```

```
4                                   0                 ?                NO

  police_report_available  fraud_reported
0                     YES               0
1                       ?               0
2                       ?               0
3                      NO               0
4                     YES               0
```

[ ]:
```
X = dummies.iloc[:, 0:-1]  # predictor variables
y = dummies.iloc[:, -1]  # target variable

len(X.columns)
```

[ ]:
```
148
```

[ ]:
```
X.head(2)
```

[ ]:
```
   policy_state_IL  policy_state_IN  policy_state_OH  insured_sex_FEMALE  \
0                0                1                0                   1
1                1                0                0                   1

   insured_sex_MALE  insured_education_level_Associate  \
0                 0                                  1
1                 0                                  0

   insured_education_level_College  insured_education_level_High School  \
0                                0                                     0
1                                0                                     0

   insured_education_level_JD  insured_education_level_MD  \
0                           0                           0
1                           0                           1

   insured_education_level_Masters  insured_education_level_PhD  \
0                                0                            0
1                                0                            0

   insured_occupation_adm-clerical  insured_occupation_armed-forces  \
0                                0                                0
1                                0                                0

   insured_occupation_craft-repair  insured_occupation_exec-managerial  \
0                                0                                   0
1                                0                                   1

   insured_occupation_farming-fishing  insured_occupation_handlers-cleaners  \
```

```
0                                   0                                        0
1                                   0                                        0

   insured_occupation_machine-op-inspct  insured_occupation_other-service  \
0                                     0                                   0
1                                     0                                   0

   insured_occupation_priv-house-serv  insured_occupation_prof-specialty  \
0                                   1                                   0
1                                   0                                   0

   insured_occupation_protective-serv  insured_occupation_sales  \
0                                   0                         0
1                                   0                         0

   insured_occupation_tech-support  insured_occupation_transport-moving  \
0                                0                                    0
1                                0                                    0

   insured_hobbies_base-jumping  insured_hobbies_basketball  \
0                             0                           0
1                             0                           0

   insured_hobbies_board-games  insured_hobbies_bungie-jumping  \
0                            0                               0
1                            0                               0

   insured_hobbies_camping  insured_hobbies_chess  insured_hobbies_cross-fit  \
0                        0                      0                          0
1                        1                      0                          0

   insured_hobbies_dancing  insured_hobbies_exercise  insured_hobbies_golf  \
0                        0                         0                     0
1                        0                         0                     0

   insured_hobbies_hiking  insured_hobbies_kayaking  insured_hobbies_movies  \
0                       0                         0                       1
1                       0                         0                       0

   insured_hobbies_paintball  insured_hobbies_polo  insured_hobbies_reading  \
0                          0                     0                        0
1                          0                     0                        0

   insured_hobbies_skydiving  insured_hobbies_sleeping  \
0                          0                         0
1                          0                         0
```

```
   insured_hobbies_video-games  insured_hobbies_yachting  \
0                            0                         0
1                            0                         0

   insured_relationship_husband  insured_relationship_not-in-family  \
0                             1                                   0
1                             0                                   0

   insured_relationship_other-relative  insured_relationship_own-child  \
0                                    0                               0
1                                    1                               0

   insured_relationship_unmarried  insured_relationship_wife  \
0                               0                          0
1                               0                          0

   incident_type_Multi-vehicle Collision  incident_type_Parked Car  \
0                                      0                         0
1                                      1                         0

   incident_type_Single Vehicle Collision  incident_type_Vehicle Theft  \
0                                       1                           0
1                                       0                           0

   incident_severity_Major Damage  incident_severity_Minor Damage  \
0                               0                               1
1                               0                               0

   incident_severity_Total Loss  incident_severity_Trivial Damage  \
0                             0                                 0
1                             1                                 0

   authorities_contacted_Ambulance  authorities_contacted_Fire  \
0                                0                           0
1                                0                           0

   authorities_contacted_None  authorities_contacted_Other  \
0                           0                             1
1                           0                             1

   authorities_contacted_Police  incident_state_NC  incident_state_NY  \
0                             0                  0                  1
1                             0                  0                  0

   incident_state_OH  incident_state_PA  incident_state_SC  incident_state_VA  \
0                  0                  0                  0                  0
1                  0                  0                  1                  0
```

```
   incident_state_WV  incident_city_Arlington  incident_city_Columbus  \
0                  0                        0                       0
1                  0                        1                       0

   incident_city_Hillsdale  incident_city_Northbend  incident_city_Northbrook  \
0                        1                       0                         0
1                        0                       0                         0

   incident_city_Riverwood  incident_city_Springfield  auto_make_Accura  \
0                        0                          0                 0
1                        0                          0                 0

   auto_make_Audi  auto_make_BMW  auto_make_Chevrolet  auto_make_Dodge  \
0               0              0                    0                0
1               0              0                    0                0

   auto_make_Ford  auto_make_Honda  auto_make_Jeep  auto_make_Mercedes  \
0               0                0               0                   0
1               0                0               0                   0

   auto_make_Nissan  auto_make_Saab  auto_make_Suburu  auto_make_Toyota  \
0                 0                1                 0                 0
1                 0                1                 0                 0

   auto_make_Volkswagen  auto_model_3 Series  auto_model_92x  auto_model_93  \
0                     0                    0               1              0
1                     0                    0               0              0

   auto_model_95  auto_model_A3  auto_model_A5  auto_model_Accord  \
0              0              0              0                  0
1              1              0              0                  0

   auto_model_C300  auto_model_CRV  auto_model_Camry  auto_model_Civic  \
0                0               0                 0                 0
1                0               0                 0                 0

   auto_model_Corolla  auto_model_E400  auto_model_Escape  auto_model_F150  \
0                   0                0                  0                0
1                   0                0                  0                0

   auto_model_Forrestor  auto_model_Fusion  auto_model_Grand Cherokee  \
0                     0                  0                          0
1                     0                  0                          0

   auto_model_Highlander  auto_model_Impreza  auto_model_Jetta  \
0                      0                   0                 0
```

```
1                      0                   0                 0

   auto_model_Legacy  auto_model_M5  auto_model_MDX  auto_model_ML350  \
0                  0              0               0                 0
1                  0              0               0                 0

   auto_model_Malibu  auto_model_Maxima  auto_model_Neon  auto_model_Passat  \
0                  0                   0                0                  0
1                  0                   0                0                  0

   auto_model_Pathfinder  auto_model_RAM  auto_model_RSX  \
0                      0               0               0
1                      0               0               0

   auto_model_Silverado  auto_model_TL  auto_model_Tahoe  auto_model_Ultima  \
0                     0              0                 0                  0
1                     0              0                 0                  0

   auto_model_Wrangler  auto_model_X5  auto_model_X6  csl_per_person_100  \
0                    0              0              0                   0
1                    0              0              0                   0

   csl_per_person_250  csl_per_person_500  csl_per_accident_1000  \
0                   1                   0                      0
1                   0                   1                      1

   csl_per_accident_300  csl_per_accident_500  \
0                     0                     1
1                     0                     0

   incident_period_of_day_afternoon  incident_period_of_day_early_morning  \
0                                 0                                     0
1                                 0                                     0

   incident_period_of_day_evening  incident_period_of_day_fore-noon  \
0                               1                                 0
1                               0                                 1

   incident_period_of_day_morning  incident_period_of_day_night  \
0                               0                             0
1                               0                             0

   incident_period_of_day_past_midnight   collision_type property_damage  \
0                                     0  Front Collision               ?
1                                     0   Rear Collision               ?

  police_report_available
```

```
0                     YES
1                       ?
```

[ ]:
```
y.head()
```

[ ]:
```
0    0
1    0
2    0
3    0
4    0
Name: fraud_reported, dtype: int64
```

4 Label encoding

[ ]:
```
from sklearn.preprocessing import LabelEncoder
X['collision_en'] = LabelEncoder().fit_transform(dummies['collision_type'])
X[['collision_type', 'collision_en']]
```

[ ]:
```
         collision_type  collision_en
0       Front Collision             1
1        Rear Collision             2
2                     ?             0
3                     ?             0
4                     ?             0
…                   …             …
10206    Rear Collision             2
10207    Rear Collision             2
10208                 ?             0
10209   Front Collision             1
10210    Side Collision             3

[10211 rows x 2 columns]
```

[ ]:
```
X['property_damage'].replace(to_replace='YES', value=1, inplace=True)
X['property_damage'].replace(to_replace='NO', value=0, inplace=True)
X['property_damage'].replace(to_replace='?', value=0, inplace=True)
X['police_report_available'].replace(to_replace='YES', value=1, inplace=True)
X['police_report_available'].replace(to_replace='NO', value=0, inplace=True)
X['police_report_available'].replace(to_replace='?', value=0, inplace=True)

X.head(10)
```

[ ]:
```
   policy_state_IL  policy_state_IN  policy_state_OH  insured_sex_FEMALE  \
0                0                1                0                   1
1                1                0                0                   1
```

```
2                 0                 0                 1                   1
3                 0                 0                 1                   1
4                 1                 0                 0                   1
5                 0                 1                 0                   1
6                 1                 0                 0                   0
7                 0                 1                 0                   0
8                 1                 0                 0                   0
9                 1                 0                 0                   0

   insured_sex_MALE  insured_education_level_Associate  \
0                 0                                  1
1                 0                                  0
2                 0                                  0
3                 0                                  0
4                 0                                  0
5                 0                                  0
6                 1                                  0
7                 1                                  0
8                 1                                  1
9                 1                                  1

   insured_education_level_College  insured_education_level_High School  \
0                                0                                    0
1                                0                                    0
2                                0                                    1
3                                0                                    1
4                                0                                    0
5                                0                                    1
6                                0                                    0
7                                0                                    0
8                                0                                    0
9                                0                                    0

   insured_education_level_JD  insured_education_level_MD  \
0                           0                           0
1                           0                           1
2                           0                           0
3                           0                           0
4                           0                           0
5                           0                           0
6                           0                           1
7                           1                           0
8                           0                           0
9                           0                           0

   insured_education_level_Masters  insured_education_level_PhD  \
0                                0                            0
```

```
1                                  0                                  0
2                                  0                                  0
3                                  0                                  0
4                                  0                                  1
5                                  0                                  0
6                                  0                                  0
7                                  0                                  0
8                                  0                                  0
9                                  0                                  0

   insured_occupation_adm-clerical  insured_occupation_armed-forces  \
0                                  0                                  0
1                                  0                                  0
2                                  0                                  0
3                                  0                                  0
4                                  0                                  1
5                                  0                                  1
6                                  0                                  0
7                                  0                                  0
8                                  0                                  0
9                                  0                                  0

   insured_occupation_craft-repair  insured_occupation_exec-managerial  \
0                                  0                                  0
1                                  0                                  1
2                                  0                                  0
3                                  0                                  0
4                                  0                                  0
5                                  0                                  0
6                                  0                                  0
7                                  0                                  0
8                                  0                                  0
9                                  0                                  0

   insured_occupation_farming-fishing  insured_occupation_handlers-cleaners  \
0                                   0                                   0
1                                   0                                   0
2                                   1                                   0
3                                   0                                   0
4                                   0                                   0
5                                   0                                   0
6                                   1                                   0
7                                   0                                   0
8                                   0                                   0
9                                   0                                   0

   insured_occupation_machine-op-inspct  insured_occupation_other-service  \
```

0	0	0
1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0

	insured_occupation_priv-house-serv	insured_occupation_prof-specialty	\
0	1	0	
1	0	0	
2	0	0	
3	0	0	
4	0	0	
5	0	0	
6	0	0	
7	0	0	
8	0	0	
9	0	0	

	insured_occupation_protective-serv	insured_occupation_sales	\
0	0	0	
1	0	0	
2	0	0	
3	0	0	
4	0	0	
5	0	0	
6	0	0	
7	0	0	
8	0	0	
9	1	0	

	insured_occupation_tech-support	insured_occupation_transport-moving	\
0	0	0	
1	0	0	
2	0	0	
3	0	1	
4	0	0	
5	0	0	
6	0	0	
7	0	1	
8	1	0	
9	0	0	

```
   insured_hobbies_base-jumping  insured_hobbies_basketball  \
0                             0                           0
1                             0                           0
2                             0                           0
3                             0                           0
4                             0                           0
5                             0                           0
6                             0                           1
7                             0                           0
8                             0                           0
9                             0                           0

   insured_hobbies_board-games  insured_hobbies_bungie-jumping  \
0                            0                               0
1                            0                               0
2                            0                               0
3                            0                               1
4                            0                               0
5                            0                               0
6                            0                               0
7                            0                               0
8                            0                               0
9                            0                               0

   insured_hobbies_camping  insured_hobbies_chess  insured_hobbies_cross-fit  \
0                        0                      0                          0
1                        1                      0                          0
2                        0                      0                          0
3                        0                      0                          0
4                        0                      0                          0
5                        0                      0                          0
6                        0                      0                          0
7                        0                      0                          0
8                        0                      0                          0
9                        0                      0                          0

   insured_hobbies_dancing  insured_hobbies_exercise  insured_hobbies_golf  \
0                        0                         0                     0
1                        0                         0                     0
2                        0                         0                     1
3                        0                         0                     0
4                        0                         1                     0
5                        1                         0                     0
6                        0                         0                     0
7                        0                         0                     0
8                        0                         0                     0
9                        0                         0                     0
```

```
   insured_hobbies_hiking  insured_hobbies_kayaking  insured_hobbies_movies  \
0                       0                         0                       1
1                       0                         0                       0
2                       0                         0                       0
3                       0                         0                       0
4                       0                         0                       0
5                       0                         0                       0
6                       0                         0                       0
7                       1                         0                       0
8                       0                         1                       0
9                       0                         0                       1

   insured_hobbies_paintball  insured_hobbies_polo  insured_hobbies_reading  \
0                          0                     0                        0
1                          0                     0                        0
2                          0                     0                        0
3                          0                     0                        0
4                          0                     0                        0
5                          0                     0                        0
6                          0                     0                        0
7                          0                     0                        0
8                          0                     0                        0
9                          0                     0                        0

   insured_hobbies_skydiving  insured_hobbies_sleeping  \
0                          0                         0
1                          0                         0
2                          0                         0
3                          0                         0
4                          0                         0
5                          0                         0
6                          0                         0
7                          0                         0
8                          0                         0
9                          0                         0

   insured_hobbies_video-games  insured_hobbies_yachting  \
0                            0                         0
1                            0                         0
2                            0                         0
3                            0                         0
4                            0                         0
5                            0                         0
6                            0                         0
7                            0                         0
8                            0                         0
```

```
9                              0                           0

   insured_relationship_husband  insured_relationship_not-in-family  \
0                             1                                   0
1                             0                                   0
2                             0                                   0
3                             1                                   0
4                             1                                   0
5                             0                                   0
6                             0                                   1
7                             0                                   0
8                             0                                   1
9                             1                                   0

   insured_relationship_other-relative  insured_relationship_own-child  \
0                                    0                               0
1                                    1                               0
2                                    0                               1
3                                    0                               0
4                                    0                               0
5                                    0                               0
6                                    0                               0
7                                    0                               1
8                                    0                               0
9                                    0                               0

   insured_relationship_unmarried  insured_relationship_wife  \
0                               0                          0
1                               0                          0
2                               0                          0
3                               0                          0
4                               0                          0
5                               0                          1
6                               0                          0
7                               0                          0
8                               0                          0
9                               0                          0

   incident_type_Multi-vehicle Collision  incident_type_Parked Car  \
0                                      0                         0
1                                      1                         0
2                                      0                         1
3                                      0                         0
4                                      0                         0
5                                      0                         0
6                                      1                         0
7                                      0                         0
```

```
8                                     0                          0
9                                     1                          0

   incident_type_Single Vehicle Collision  incident_type_Vehicle Theft  \
0                                       1                           0
1                                       0                           0
2                                       0                           0
3                                       0                           1
4                                       0                           1
5                                       1                           0
6                                       0                           0
7                                       1                           0
8                                       1                           0
9                                       0                           0

   incident_severity_Major Damage  incident_severity_Minor Damage  \
0                               0                               1
1                               0                               0
2                               0                               0
3                               0                               0
4                               0                               1
5                               0                               1
6                               0                               0
7                               0                               0
8                               0                               1
9                               1                               0

   incident_severity_Total Loss  incident_severity_Trivial Damage  \
0                             0                                 0
1                             1                                 0
2                             0                                 1
3                             0                                 1
4                             0                                 0
5                             0                                 0
6                             1                                 0
7                             1                                 0
8                             0                                 0
9                             0                                 0

   authorities_contacted_Ambulance  authorities_contacted_Fire  \
0                                0                           0
1                                0                           0
2                                0                           0
3                                0                           0
4                                0                           0
5                                0                           1
6                                0                           0
```

```
7                                0                           1
8                                1                           0
9                                0                           0

   authorities_contacted_None  authorities_contacted_Other  \
0                           0                            1
1                           0                            1
2                           1                            0
3                           0                            0
4                           1                            0
5                           0                            0
6                           0                            1
7                           0                            0
8                           0                            0
9                           0                            1

   authorities_contacted_Police  incident_state_NC  incident_state_NY  \
0                             0                  0                  1
1                             0                  0                  0
2                             0                  0                  0
3                             1                  0                  0
4                             0                  0                  0
5                             0                  0                  1
6                             0                  0                  0
7                             0                  0                  1
8                             0                  0                  0
9                             0                  1                  0

   incident_state_OH  incident_state_PA  incident_state_SC  incident_state_VA  \
0                  0                  0                  0                  0
1                  0                  0                  1                  0
2                  0                  0                  1                  0
3                  0                  0                  0                  0
4                  1                  0                  0                  0
5                  0                  0                  0                  0
6                  0                  0                  0                  1
7                  0                  0                  0                  0
8                  0                  0                  0                  0
9                  0                  0                  0                  0

   incident_state_WV  incident_city_Arlington  incident_city_Columbus  \
0                  0                        0                       0
1                  0                        1                       0
2                  0                        1                       0
3                  1                        0                       0
4                  0                        0                       0
5                  0                        0                       1
```

```
6                    0                         0                         0
7                    0                         0                         0
8                    1                         0                         1
9                    0                         1                         0

   incident_city_Hillsdale  incident_city_Northbend  incident_city_Northbrook  \
0                        1                       0                         0
1                        0                       0                         0
2                        0                       0                         0
3                        0                       0                         0
4                        0                       0                         1
5                        0                       0                         0
6                        0                       0                         1
7                        0                       0                         0
8                        0                       0                         0
9                        0                       0                         0

   incident_city_Riverwood  incident_city_Springfield  auto_make_Accura  \
0                        0                         0                 0
1                        0                         0                 0
2                        0                         0                 0
3                        0                         1                 0
4                        0                         0                 0
5                        0                         0                 0
6                        0                         0                 0
7                        1                         0                 1
8                        0                         0                 0
9                        0                         0                 0

   auto_make_Audi  auto_make_BMW  auto_make_Chevrolet  auto_make_Dodge  \
0               0              0                    0                0
1               0              0                    0                0
2               0              0                    1                0
3               0              0                    0                0
4               0              0                    0                0
5               0              0                    0                0
6               0              0                    1                0
7               0              0                    0                0
8               0              1                    0                0
9               1              0                    0                0

   auto_make_Ford  auto_make_Honda  auto_make_Jeep  auto_make_Mercedes  \
0               0                0               0                   0
1               0                0               0                   0
2               0                0               0                   0
3               0                0               0                   0
4               0                0               0                   0
```

```
5                0                1                0                    0
6                0                0                0                    0
7                0                0                0                    0
8                0                0                0                    0
9                0                0                0                    0

   auto_make_Nissan  auto_make_Saab  auto_make_Suburu  auto_make_Toyota  \
0                 0                1                 0                 0
1                 0                1                 0                 0
2                 0                0                 0                 0
3                 0                1                 0                 0
4                 0                0                 0                 1
5                 0                0                 0                 0
6                 0                0                 0                 0
7                 0                0                 0                 0
8                 0                0                 0                 0
9                 0                0                 0                 0

   auto_make_Volkswagen  auto_model_3 Series  auto_model_92x  auto_model_93  \
0                     0                    0               1              0
1                     0                    0               0              0
2                     0                    0               0              0
3                     0                    0               0              0
4                     0                    0               0              0
5                     0                    0               0              0
6                     0                    0               0              0
7                     0                    0               0              0
8                     0                    0               0              0
9                     0                    0               0              0

   auto_model_95  auto_model_A3  auto_model_A5  auto_model_Accord  \
0              0              0              0                  0
1              1              0              0                  0
2              0              0              0                  0
3              1              0              0                  0
4              0              0              0                  0
5              0              0              0                  0
6              0              0              0                  0
7              0              0              0                  0
8              0              0              0                  0
9              0              0              1                  0

   auto_model_C300  auto_model_CRV  auto_model_Camry  auto_model_Civic  \
0                0               0                 0                 0
1                0               0                 0                 0
2                0               0                 0                 0
3                0               0                 0                 0
```

```
4                0              0                0                0
5                0              0                0                1
6                0              0                0                0
7                0              0                0                0
8                0              0                0                0
9                0              0                0                0

   auto_model_Corolla  auto_model_E400  auto_model_Escape  auto_model_F150  \
0                   0                0                  0                0
1                   0                0                  0                0
2                   0                0                  0                0
3                   0                0                  0                0
4                   0                0                  0                0
5                   0                0                  0                0
6                   0                0                  0                0
7                   0                0                  0                0
8                   0                0                  0                0
9                   0                0                  0                0

   auto_model_Forrestor  auto_model_Fusion  auto_model_Grand Cherokee  \
0                     0                  0                          0
1                     0                  0                          0
2                     0                  0                          0
3                     0                  0                          0
4                     0                  0                          0
5                     0                  0                          0
6                     0                  0                          0
7                     0                  0                          0
8                     0                  0                          0
9                     0                  0                          0

   auto_model_Highlander  auto_model_Impreza  auto_model_Jetta  \
0                      0                   0                 0
1                      0                   0                 0
2                      0                   0                 0
3                      0                   0                 0
4                      1                   0                 0
5                      0                   0                 0
6                      0                   0                 0
7                      0                   0                 0
8                      0                   0                 0
9                      0                   0                 0

   auto_model_Legacy  auto_model_M5  auto_model_MDX  auto_model_ML350  \
0                  0              0               0                 0
1                  0              0               0                 0
2                  0              0               0                 0
```

```
3                   0              0               0                   0
4                   0              0               0                   0
5                   0              0               0                   0
6                   0              0               0                   0
7                   0              0               0                   0
8                   0              1               0                   0
9                   0              0               0                   0

   auto_model_Malibu  auto_model_Maxima  auto_model_Neon  auto_model_Passat  \
0                  0                  0                0                  0
1                  0                  0                0                  0
2                  1                  0                0                  0
3                  0                  0                0                  0
4                  0                  0                0                  0
5                  0                  0                0                  0
6                  0                  0                0                  0
7                  0                  0                0                  0
8                  0                  0                0                  0
9                  0                  0                0                  0

   auto_model_Pathfinder  auto_model_RAM  auto_model_RSX  \
0                      0               0               0
1                      0               0               0
2                      0               0               0
3                      0               0               0
4                      0               0               0
5                      0               0               0
6                      0               0               0
7                      0               0               1
8                      0               0               0
9                      0               0               0

   auto_model_Silverado  auto_model_TL  auto_model_Tahoe  auto_model_Ultima  \
0                     0              0                 0                  0
1                     0              0                 0                  0
2                     0              0                 0                  0
3                     0              0                 0                  0
4                     0              0                 0                  0
5                     0              0                 0                  0
6                     0              0                 1                  0
7                     0              0                 0                  0
8                     0              0                 0                  0
9                     0              0                 0                  0

   auto_model_Wrangler  auto_model_X5  auto_model_X6  csl_per_person_100  \
0                    0              0              0                   0
1                    0              0              0                   0
```

```
2                    0              0              0                    1
3                    0              0              0                    1
4                    0              0              0                    1
5                    0              0              0                    1
6                    0              0              0                    0
7                    0              0              0                    1
8                    0              0              0                    1
9                    0              0              0                    0

   csl_per_person_250  csl_per_person_500  csl_per_accident_1000  \
0                   1                   0                      0
1                   0                   1                      1
2                   0                   0                      0
3                   0                   0                      0
4                   0                   0                      0
5                   0                   0                      0
6                   1                   0                      0
7                   0                   0                      0
8                   0                   0                      0
9                   1                   0                      0

   csl_per_accident_300  csl_per_accident_500  \
0                     0                     1
1                     0                     0
2                     1                     0
3                     1                     0
4                     1                     0
5                     1                     0
6                     0                     1
7                     1                     0
8                     1                     0
9                     0                     1

   incident_period_of_day_afternoon  incident_period_of_day_early_morning  \
0                                 0                                     0
1                                 0                                     0
2                                 0                                     0
3                                 0                                     1
4                                 0                                     1
5                                 0                                     1
6                                 0                                     0
7                                 0                                     0
8                                 0                                     0
9                                 1                                     0

   incident_period_of_day_evening  incident_period_of_day_fore-noon  \
0                               1                                 0
```

```
1                                  0                                  1
2                                  0                                  0
3                                  0                                  0
4                                  0                                  0
5                                  0                                  0
6                                  0                                  1
7                                  0                                  0
8                                  0                                  0
9                                  0                                  0

   incident_period_of_day_morning  incident_period_of_day_night  \
0                               0                            0
1                               0                            0
2                               0                            0
3                               0                            0
4                               0                            0
5                               0                            0
6                               0                            0
7                               0                            0
8                               0                            1
9                               0                            0

   incident_period_of_day_past_midnight   collision_type  property_damage  \
0                                     0  Front Collision                0
1                                     0   Rear Collision                0
2                                     1                ?                0
3                                     0                ?                1
4                                     0                ?                0
5                                     0   Rear Collision                1
6                                     0   Rear Collision                0
7                                     1  Front Collision                0
8                                     0   Rear Collision                0
9                                     0   Side Collision                1

   police_report_available  collision_en
0                        1             1
1                        0             2
2                        0             0
3                        0             0
4                        1             0
5                        0             2
6                        1             2
7                        0             1
8                        0             2
9                        0             3
```

[ ]:
```
X = X.drop(columns = ['collision_type'])
X.head(2)
```

[ ]:

	policy_state_IL	policy_state_IN	policy_state_OH	insured_sex_FEMALE	\
0	0	1	0	1	
1	1	0	0	1	

	insured_sex_MALE	insured_education_level_Associate	\
0	0	1	
1	0	0	

	insured_education_level_College	insured_education_level_High School	\
0	0	0	
1	0	0	

	insured_education_level_JD	insured_education_level_MD	\
0	0	0	
1	0	1	

	insured_education_level_Masters	insured_education_level_PhD	\
0	0	0	
1	0	0	

	insured_occupation_adm-clerical	insured_occupation_armed-forces	\
0	0	0	
1	0	0	

	insured_occupation_craft-repair	insured_occupation_exec-managerial	\
0	0	0	
1	0	1	

	insured_occupation_farming-fishing	insured_occupation_handlers-cleaners	\
0	0	0	
1	0	0	

	insured_occupation_machine-op-inspct	insured_occupation_other-service	\
0	0	0	
1	0	0	

	insured_occupation_priv-house-serv	insured_occupation_prof-specialty	\
0	1	0	
1	0	0	

	insured_occupation_protective-serv	insured_occupation_sales	\
0	0	0	
1	0	0	

```
   insured_occupation_tech-support  insured_occupation_transport-moving  \
0                                0                                    0
1                                0                                    0

   insured_hobbies_base-jumping  insured_hobbies_basketball  \
0                             0                           0
1                             0                           0

   insured_hobbies_board-games  insured_hobbies_bungie-jumping  \
0                            0                               0
1                            0                               0

   insured_hobbies_camping  insured_hobbies_chess  insured_hobbies_cross-fit  \
0                        0                      0                          0
1                        1                      0                          0

   insured_hobbies_dancing  insured_hobbies_exercise  insured_hobbies_golf  \
0                        0                         0                     0
1                        0                         0                     0

   insured_hobbies_hiking  insured_hobbies_kayaking  insured_hobbies_movies  \
0                       0                         0                       1
1                       0                         0                       0

   insured_hobbies_paintball  insured_hobbies_polo  insured_hobbies_reading  \
0                          0                     0                        0
1                          0                     0                        0

   insured_hobbies_skydiving  insured_hobbies_sleeping  \
0                          0                         0
1                          0                         0

   insured_hobbies_video-games  insured_hobbies_yachting  \
0                            0                         0
1                            0                         0

   insured_relationship_husband  insured_relationship_not-in-family  \
0                             1                                   0
1                             0                                   0

   insured_relationship_other-relative  insured_relationship_own-child  \
0                                    0                               0
1                                    1                               0

   insured_relationship_unmarried  insured_relationship_wife  \
0                               0                          0
1                               0                          0
```

```
   incident_type_Multi-vehicle Collision  incident_type_Parked Car  \
0                                      0                         0
1                                      1                         0

   incident_type_Single Vehicle Collision  incident_type_Vehicle Theft  \
0                                       1                           0
1                                       0                           0

   incident_severity_Major Damage  incident_severity_Minor Damage  \
0                               0                               1
1                               0                               0

   incident_severity_Total Loss  incident_severity_Trivial Damage  \
0                             0                                 0
1                             1                                 0

   authorities_contacted_Ambulance  authorities_contacted_Fire  \
0                                0                          0
1                                0                          0

   authorities_contacted_None  authorities_contacted_Other  \
0                           0                            1
1                           0                            1

   authorities_contacted_Police  incident_state_NC  incident_state_NY  \
0                             0                 0                  1
1                             0                 0                  0

   incident_state_OH  incident_state_PA  incident_state_SC  incident_state_VA  \
0                  0                  0                  0                  0
1                  0                  0                  1                  0

   incident_state_WV  incident_city_Arlington  incident_city_Columbus  \
0                  0                        0                       0
1                  0                        1                       0

   incident_city_Hillsdale  incident_city_Northbend  incident_city_Northbrook  \
0                        1                        0                         0
1                        0                        0                         0

   incident_city_Riverwood  incident_city_Springfield  auto_make_Accura  \
0                        0                          0                 0
1                        0                          0                 0

   auto_make_Audi  auto_make_BMW  auto_make_Chevrolet  auto_make_Dodge  \
0               0               0                    0                0
```

```
1               0              0                    0                0

   auto_make_Ford  auto_make_Honda  auto_make_Jeep  auto_make_Mercedes  \
0               0                0               0                   0
1               0                0               0                   0

   auto_make_Nissan  auto_make_Saab  auto_make_Suburu  auto_make_Toyota  \
0                 0               1                 0                 0
1                 0               1                 0                 0

   auto_make_Volkswagen  auto_model_3 Series  auto_model_92x  auto_model_93  \
0                     0                    0               1              0
1                     0                    0               0              0

   auto_model_95  auto_model_A3  auto_model_A5  auto_model_Accord  \
0              0              0              0                  0
1              1              0              0                  0

   auto_model_C300  auto_model_CRV  auto_model_Camry  auto_model_Civic  \
0                0               0                 0                 0
1                0               0                 0                 0

   auto_model_Corolla  auto_model_E400  auto_model_Escape  auto_model_F150  \
0                   0                0                  0                0
1                   0                0                  0                0

   auto_model_Forrestor  auto_model_Fusion  auto_model_Grand Cherokee  \
0                     0                  0                          0
1                     0                  0                          0

   auto_model_Highlander  auto_model_Impreza  auto_model_Jetta  \
0                      0                   0                 0
1                      0                   0                 0

   auto_model_Legacy  auto_model_M5  auto_model_MDX  auto_model_ML350  \
0                  0              0               0                 0
1                  0              0               0                 0

   auto_model_Malibu  auto_model_Maxima  auto_model_Neon  auto_model_Passat  \
0                  0                  0                0                  0
1                  0                  0                0                  0

   auto_model_Pathfinder  auto_model_RAM  auto_model_RSX  \
0                      0               0               0
1                      0               0               0

   auto_model_Silverado  auto_model_TL  auto_model_Tahoe  auto_model_Ultima  \
```

```
0                     0              0                 0                   0
1                     0              0                 0                   0

   auto_model_Wrangler  auto_model_X5  auto_model_X6  csl_per_person_100  \
0                    0              0              0                   0
1                    0              0              0                   0

   csl_per_person_250  csl_per_person_500  csl_per_accident_1000  \
0                   1                   0                      0
1                   0                   1                      1

   csl_per_accident_300  csl_per_accident_500  \
0                     0                     1
1                     0                     0

   incident_period_of_day_afternoon  incident_period_of_day_early_morning  \
0                                 0                                     0
1                                 0                                     0

   incident_period_of_day_evening  incident_period_of_day_fore-noon  \
0                               1                                 0
1                               0                                 1

   incident_period_of_day_morning  incident_period_of_day_night  \
0                               0                             0
1                               0                             0

   incident_period_of_day_past_midnight  property_damage  \
0                                     0               0
1                                     0               0

   police_report_available  collision_en
0                        1            1
1                        0            2
```

[ ]:
```
X = pd.concat([X, df._get_numeric_data()], axis=1)  # joining numeric columns
X.head(2)
```

[ ]:
```
   policy_state_IL  policy_state_IN  policy_state_OH  insured_sex_FEMALE  \
0                0                1                0                   1
1                1                0                0                   1

   insured_sex_MALE  insured_education_level_Associate  \
0                 0                                  1
1                 0                                  0

   insured_education_level_College  insured_education_level_High School  \
```

```
0                                0                                      0
1                                0                                      0

   insured_education_level_JD  insured_education_level_MD  \
0                           0                           0
1                           0                           1

   insured_education_level_Masters  insured_education_level_PhD  \
0                                0                            0
1                                0                            0

   insured_occupation_adm-clerical  insured_occupation_armed-forces  \
0                                0                                 0
1                                0                                 0

   insured_occupation_craft-repair  insured_occupation_exec-managerial  \
0                                0                                   0
1                                0                                   1

   insured_occupation_farming-fishing  insured_occupation_handlers-cleaners  \
0                                   0                                      0
1                                   0                                      0

   insured_occupation_machine-op-inspct  insured_occupation_other-service  \
0                                     0                                  0
1                                     0                                  0

   insured_occupation_priv-house-serv  insured_occupation_prof-specialty  \
0                                   1                                   0
1                                   0                                   0

   insured_occupation_protective-serv  insured_occupation_sales  \
0                                   0                         0
1                                   0                         0

   insured_occupation_tech-support  insured_occupation_transport-moving  \
0                                0                                   0
1                                0                                   0

   insured_hobbies_base-jumping  insured_hobbies_basketball  \
0                             0                           0
1                             0                           0

   insured_hobbies_board-games  insured_hobbies_bungie-jumping  \
0                            0                               0
1                            0                               0
```

```
   insured_hobbies_camping  insured_hobbies_chess  insured_hobbies_cross-fit  \
0                        0                      0                          0
1                        1                      0                          0

   insured_hobbies_dancing  insured_hobbies_exercise  insured_hobbies_golf  \
0                        0                         0                     0
1                        0                         0                     0

   insured_hobbies_hiking  insured_hobbies_kayaking  insured_hobbies_movies  \
0                       0                         0                       1
1                       0                         0                       0

   insured_hobbies_paintball  insured_hobbies_polo  insured_hobbies_reading  \
0                          0                     0                        0
1                          0                     0                        0

   insured_hobbies_skydiving  insured_hobbies_sleeping  \
0                          0                         0
1                          0                         0

   insured_hobbies_video-games  insured_hobbies_yachting  \
0                            0                         0
1                            0                         0

   insured_relationship_husband  insured_relationship_not-in-family  \
0                             1                                   0
1                             0                                   0

   insured_relationship_other-relative  insured_relationship_own-child  \
0                                    0                               0
1                                    1                               0

   insured_relationship_unmarried  insured_relationship_wife  \
0                               0                          0
1                               0                          0

   incident_type_Multi-vehicle Collision  incident_type_Parked Car  \
0                                      0                         0
1                                      1                         0

   incident_type_Single Vehicle Collision  incident_type_Vehicle Theft  \
0                                       1                           0
1                                       0                           0

   incident_severity_Major Damage  incident_severity_Minor Damage  \
0                               0                               1
1                               0                               0
```

```
   incident_severity_Total Loss  incident_severity_Trivial Damage  \
0                             0                                 0
1                             1                                 0

   authorities_contacted_Ambulance  authorities_contacted_Fire  \
0                                0                           0
1                                0                           0

   authorities_contacted_None  authorities_contacted_Other  \
0                           0                            1
1                           0                            1

   authorities_contacted_Police  incident_state_NC  incident_state_NY  \
0                             0                 0                  1
1                             0                 0                  0

   incident_state_OH  incident_state_PA  incident_state_SC  incident_state_VA  \
0                  0                  0                  0                  0
1                  0                  0                  1                  0

   incident_state_WV  incident_city_Arlington  incident_city_Columbus  \
0                  0                        0                       0
1                  0                        1                       0

   incident_city_Hillsdale  incident_city_Northbend  incident_city_Northbrook  \
0                        1                        0                         0
1                        0                        0                         0

   incident_city_Riverwood  incident_city_Springfield  auto_make_Accura  \
0                        0                          0                 0
1                        0                          0                 0

   auto_make_Audi  auto_make_BMW  auto_make_Chevrolet  auto_make_Dodge  \
0               0              0                    0                0
1               0              0                    0                0

   auto_make_Ford  auto_make_Honda  auto_make_Jeep  auto_make_Mercedes  \
0               0                0               0                   0
1               0                0               0                   0

   auto_make_Nissan  auto_make_Saab  auto_make_Suburu  auto_make_Toyota  \
0                 0                1                 0                 0
1                 0                1                 0                 0

   auto_make_Volkswagen  auto_model_3 Series  auto_model_92x  auto_model_93  \
0                     0                    0               1              0
```

```
1                     0                    0              0              0

   auto_model_95  auto_model_A3  auto_model_A5  auto_model_Accord  \
0              0              0              0                  0
1              1              0              0                  0

   auto_model_C300  auto_model_CRV  auto_model_Camry  auto_model_Civic  \
0                0               0                 0                 0
1                0               0                 0                 0

   auto_model_Corolla  auto_model_E400  auto_model_Escape  auto_model_F150  \
0                   0                0                  0                0
1                   0                0                  0                0

   auto_model_Forrestor  auto_model_Fusion  auto_model_Grand Cherokee  \
0                     0                 0                          0
1                     0                 0                          0

   auto_model_Highlander  auto_model_Impreza  auto_model_Jetta  \
0                      0                  0                 0
1                      0                  0                 0

   auto_model_Legacy  auto_model_M5  auto_model_MDX  auto_model_ML350  \
0                  0              0               0                 0
1                  0              0               0                 0

   auto_model_Malibu  auto_model_Maxima  auto_model_Neon  auto_model_Passat  \
0                  0                  0                0                  0
1                  0                  0                0                  0

   auto_model_Pathfinder  auto_model_RAM  auto_model_RSX  \
0                      0               0               0
1                      0               0               0

   auto_model_Silverado  auto_model_TL  auto_model_Tahoe  auto_model_Ultima  \
0                     0              0                 0                  0
1                     0              0                 0                  0

   auto_model_Wrangler  auto_model_X5  auto_model_X6  csl_per_person_100  \
0                    0              0              0                   0
1                    0              0              0                   0

   csl_per_person_250  csl_per_person_500  csl_per_accident_1000  \
0                   1                   0                      0
1                   0                   1                      1

   csl_per_accident_300  csl_per_accident_500  \
```

```
0                     0                     1
1                     0                     0

   incident_period_of_day_afternoon  incident_period_of_day_early_morning  \
0                                 0                                     0
1                                 0                                     0

   incident_period_of_day_evening  incident_period_of_day_fore-noon  \
0                               1                                 0
1                               0                                 1

   incident_period_of_day_morning  incident_period_of_day_night  \
0                               0                             0
1                               0                             0

   incident_period_of_day_past_midnight  property_damage  \
0                                     0               0
1                                     0               0

   police_report_available  collision_en  months_as_customer  age  \
0                        1             1                   5   37
1                        0             2                 462   58

   policy_deductable  policy_annual_premium  umbrella_limit  capital.gains  \
0                500                1145.28             0.0          54735
1               1000                1156.80             0.0           1381

   capital.loss  number_of_vehicles_involved  bodily_injuries  witnesses  \
0         88553                           1                2          1
1         50621                           2                0          5

   total_claim_amount  injury_claim  property_claim  vehicle_claim  \
0             96200.0          3000             500          58870
1             31200.0          3830            7370          32130

   fraud_reported  vehicle_age
0               0           21
1               0           12
```

[ ]:
```
X.columns
```

[ ]:
```
Index(['policy_state_IL', 'policy_state_IN', 'policy_state_OH',
       'insured_sex_FEMALE', 'insured_sex_MALE',
       'insured_education_level_Associate', 'insured_education_level_College',
       'insured_education_level_High School', 'insured_education_level_JD',
       'insured_education_level_MD',
       …
```

```
       'capital.loss', 'number_of_vehicles_involved', 'bodily_injuries',
       'witnesses', 'total_claim_amount', 'injury_claim', 'property_claim',
       'vehicle_claim', 'fraud_reported', 'vehicle_age'],
      dtype='object', length=164)
```

[ ]:
```
X = X.drop(columns = ['fraud_reported'])  # dropping target variable␣
↪'fraud_reported'
X.columns
```

[ ]:
```
Index(['policy_state_IL', 'policy_state_IN', 'policy_state_OH',
       'insured_sex_FEMALE', 'insured_sex_MALE',
       'insured_education_level_Associate', 'insured_education_level_College',
       'insured_education_level_High School', 'insured_education_level_JD',
       'insured_education_level_MD',
       …
       'capital.gains', 'capital.loss', 'number_of_vehicles_involved',
       'bodily_injuries', 'witnesses', 'total_claim_amount', 'injury_claim',
       'property_claim', 'vehicle_claim', 'vehicle_age'],
      dtype='object', length=163)
```

5 We now have a dataset that we could use to evaluate an algorithm sensitive to missing values like LDA.

[ ]:
```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score

# evaluate an LDA model on the dataset using k-fold cross validation
model = LinearDiscriminantAnalysis()
kfold = KFold(n_splits=5, random_state=7)
result = cross_val_score(model, X, y, cv=kfold, scoring='accuracy')
print(result.mean())
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


0.8486929162094305
```

[ ]:
```
print("Accuracy: %0.2f (+/- %0.2f)" % (result.mean(), result.std() * 2))
```

```
Accuracy: 0.85 (+/- 0.03)
```

84 % cross validation score without standardizing the data. Above is the mean score and the 95% confidence interval of the score estimate. This looks good to go for other Classification methods.

Creating a Training Set for the Data Set

[ ]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, train_size=0.8,␣
 ↪random_state=7, shuffle=True)
print('length of X_train and X_test: ', len(X_train), len(X_test))
print('length of y_train and y_test: ', len(y_train), len(y_test))
```

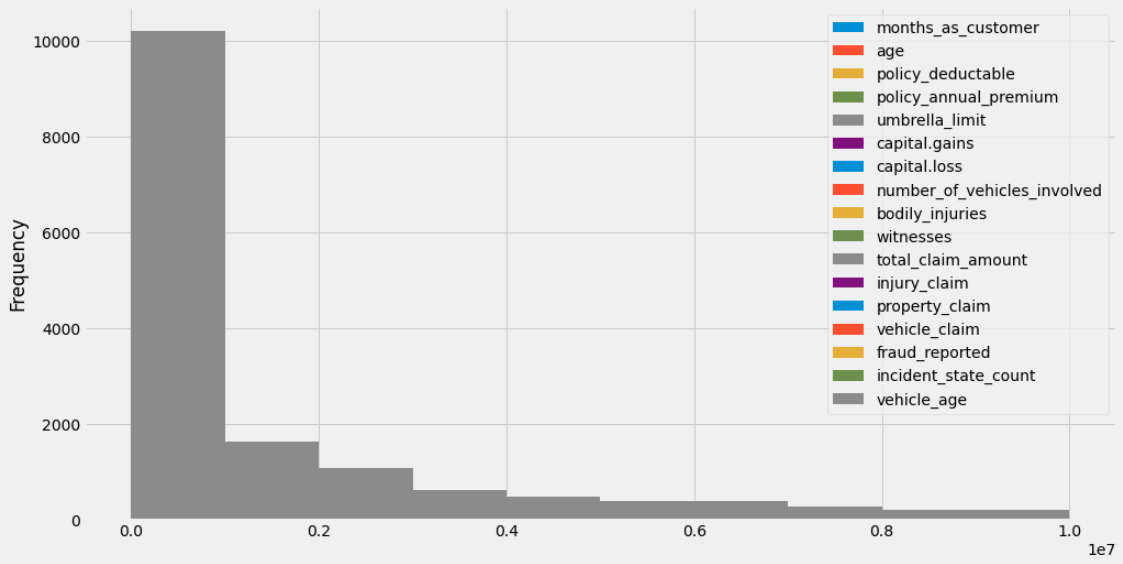
```
length of X_train and X_test:  8168 2043
length of y_train and y_test:  8168 2043
```

6 Random Forest Classification

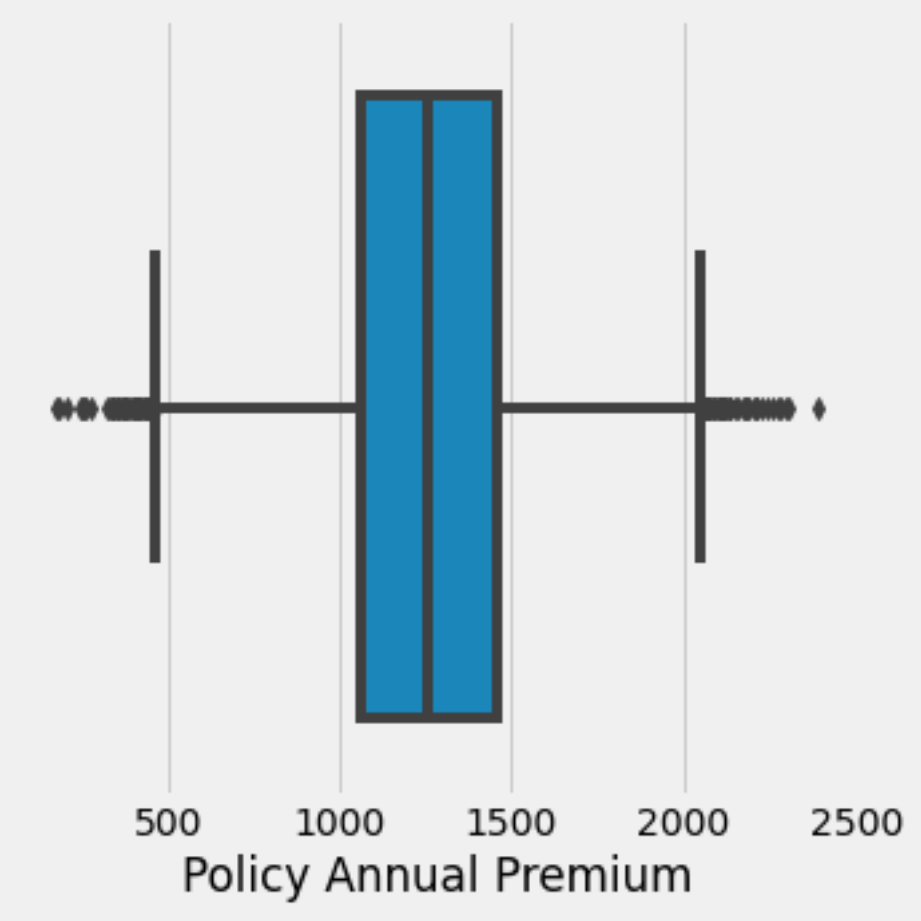
[ ]:
```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, recall_score,␣
 ↪classification_report, cohen_kappa_score
from sklearn import metrics

# Baseline Random forest based Model
rfc = RandomForestClassifier(n_estimators=200)

kfold = KFold(n_splits=5, random_state=7)
result2 = cross_val_score(rfc, X_train, y_train, cv=kfold, scoring='accuracy')
print(result2.mean())
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


0.9976738189768699
```

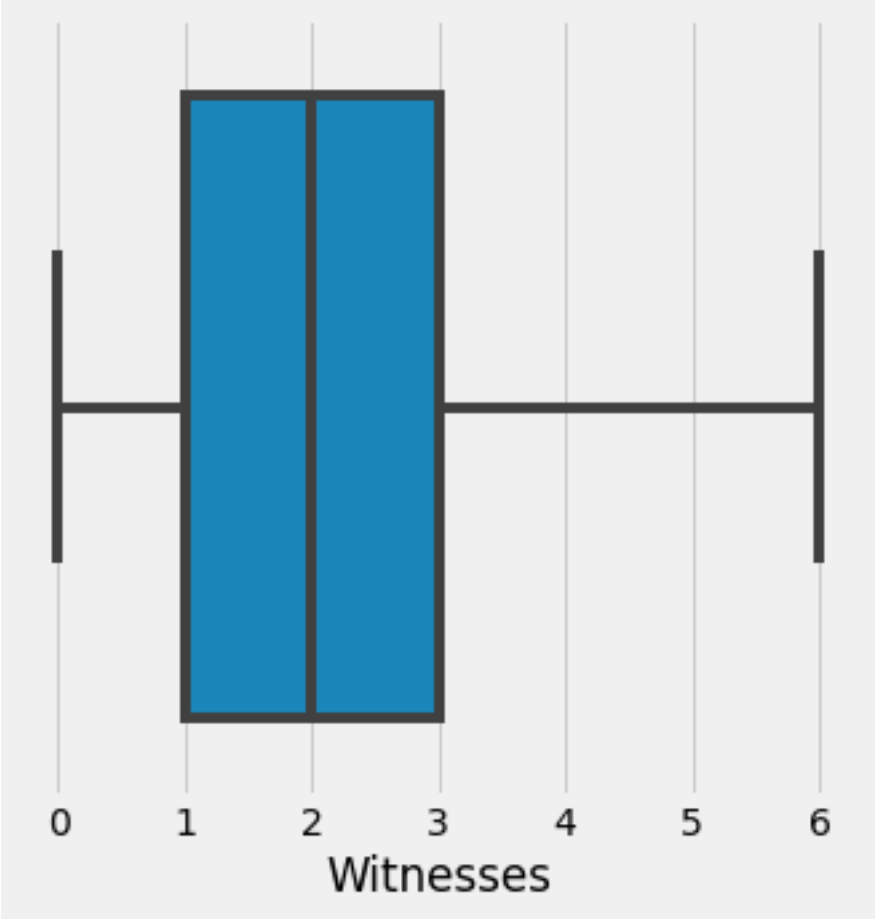
[ ]:
```
# Generate a Histogram plot for anomaly detection
plt.style.use('fivethirtyeight')
plt.rcParams['figure.figsize'] = [15, 8]
df.plot(kind='hist')
plt.show()
```



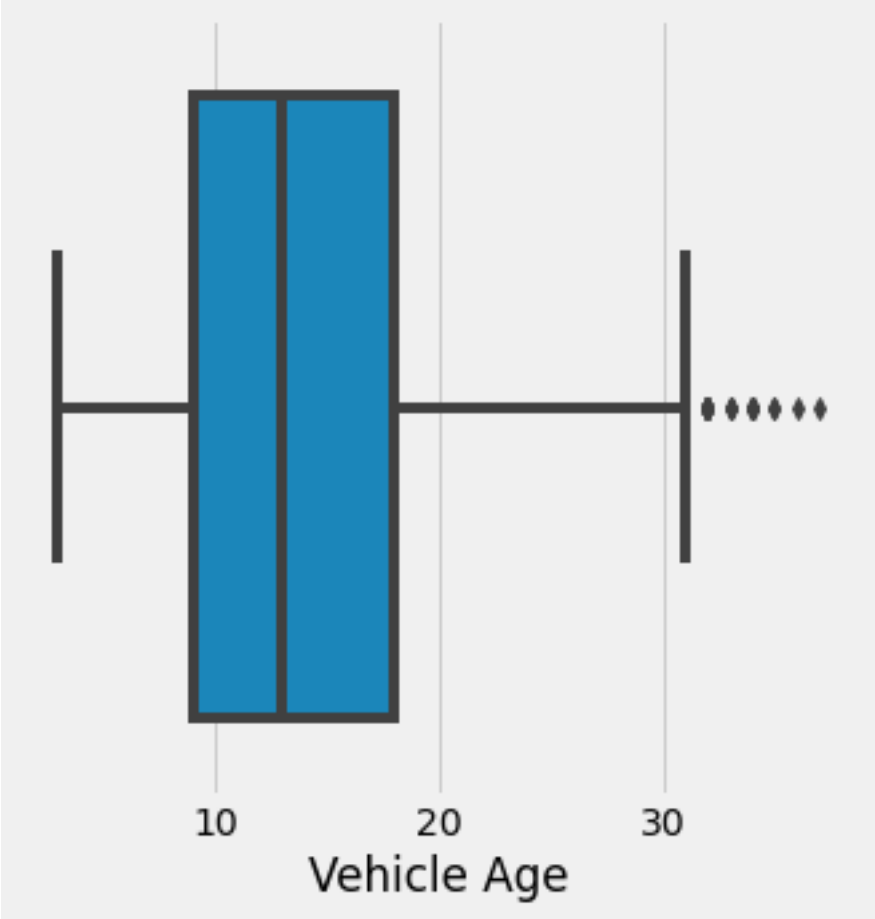
[ ]:
```
plt.rcParams['figure.figsize'] = [5, 5]
sns.boxplot(x=X.policy_annual_premium)
plt.xlabel('Policy Annual Premium')
plt.show()
```



[ ]:
```
plt.rcParams['figure.figsize'] = [5, 5]
sns.boxplot(x=X.witnesses)
plt.xlabel('Witnesses')
plt.show()
```



[ ]:
```
plt.rcParams['figure.figsize'] = [5, 5]
sns.boxplot(x=X.vehicle_age)
plt.xlabel('Vehicle Age')
plt.show()
```



[ ]:
```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler(with_mean=False)
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

X_train_scaled
```

[ ]:
```
array([[0.        , 0.        , 2.09909411, …, 2.52213226, 3.58073673,
        2.32400293],
       [0.        , 2.13021908, 0.        , …, 1.10683502, 2.79361616,
        1.39440176],
       [2.13684297, 0.        , 0.        , …, 2.26810455, 3.01916235,
        2.63386999],
       …,
       [0.        , 0.        , 2.09909411, …, 2.09391412, 3.31955118,
        1.85920235],
       [0.        , 0.        , 2.09909411, …, 2.1410907 , 2.267172  ,
        1.85920235],
```

```
       [2.13684297, 0.        , 0.        , …, 0.        , 0.98975575,
        2.47893646]])
```

[ ]:
```
X_train_scaled = pd.DataFrame(X_train_scaled, columns = X_train.columns) #␣
 ↪retaining columns names
X_train_scaled.head(2)
```

[ ]:

	policy_state_IL	policy_state_IN	policy_state_OH	insured_sex_FEMALE
0	0.0	0.000000	2.099094	0.0
1	0.0	2.130219	0.000000	0.0

	insured_sex_MALE	insured_education_level_Associate
0	2.005862	0.000000
1	2.005862	2.854538

	insured_education_level_College	insured_education_level_High School
0	0.0	0.0
1	0.0	0.0

	insured_education_level_JD	insured_education_level_MD
0	0.0	0.0
1	0.0	0.0

	insured_education_level_Masters	insured_education_level_PhD
0	2.884615	0.0
1	0.000000	0.0

	insured_occupation_adm-clerical	insured_occupation_armed-forces
0	0.0	0.0
1	0.0	0.0

	insured_occupation_craft-repair	insured_occupation_exec-managerial
0	0.0	0.0
1	0.0	0.0

	insured_occupation_farming-fishing	insured_occupation_handlers-cleaners
0	0.0	0.0
1	0.0	0.0

	insured_occupation_machine-op-inspct	insured_occupation_other-service
0	0.0	0.000000
1	0.0	4.118034

	insured_occupation_priv-house-serv	insured_occupation_prof-specialty
0	0.0	0.0
1	0.0	0.0

```
   insured_occupation_protective-serv  insured_occupation_sales  \
0                            4.179298                      0.0
1                            0.000000                      0.0

   insured_occupation_tech-support  insured_occupation_transport-moving  \
0                              0.0                                  0.0
1                              0.0                                  0.0

   insured_hobbies_base-jumping  insured_hobbies_basketball  \
0                           0.0                         0.0
1                           0.0                         0.0

   insured_hobbies_board-games  insured_hobbies_bungie-jumping  \
0                          0.0                        0.000000
1                          0.0                        4.443827

   insured_hobbies_camping  insured_hobbies_chess  insured_hobbies_cross-fit  \
0                      0.0               3.710636                        0.0
1                      0.0               0.000000                        0.0

   insured_hobbies_dancing  insured_hobbies_exercise  insured_hobbies_golf  \
0                      0.0                       0.0                   0.0
1                      0.0                       0.0                   0.0

   insured_hobbies_hiking  insured_hobbies_kayaking  insured_hobbies_movies  \
0                     0.0                       0.0                     0.0
1                     0.0                       0.0                     0.0

   insured_hobbies_paintball  insured_hobbies_polo  insured_hobbies_reading  \
0                        0.0                   0.0                      0.0
1                        0.0                   0.0                      0.0

   insured_hobbies_skydiving  insured_hobbies_sleeping  \
0                        0.0                       0.0
1                        0.0                       0.0

   insured_hobbies_video-games  insured_hobbies_yachting  \
0                          0.0                       0.0
1                          0.0                       0.0

   insured_relationship_husband  insured_relationship_not-in-family  \
0                      0.000000                                 0.0
1                      2.671049                                 0.0

   insured_relationship_other-relative  insured_relationship_own-child  \
0                                  0.0                             0.0
1                                  0.0                             0.0
```

```
   insured_relationship_unmarried  insured_relationship_wife  \
0                        2.872019                        0.0
1                        0.000000                        0.0

   incident_type_Multi-vehicle Collision  incident_type_Parked Car  \
0                               2.022539                       0.0
1                               0.000000                       0.0

   incident_type_Single Vehicle Collision  incident_type_Vehicle Theft  \
0                                0.000000                          0.0
1                                2.018796                          0.0

   incident_severity_Major Damage  incident_severity_Minor Damage  \
0                        0.000000                             0.0
1                        2.031915                             0.0

   incident_severity_Total Loss  incident_severity_Trivial Damage  \
0                      2.389131                             0.0
1                      0.000000                             0.0

   authorities_contacted_Ambulance  authorities_contacted_Fire  \
0                         0.000000                        0.0
1                         2.504959                        0.0

   authorities_contacted_None  authorities_contacted_Other  \
0                         0.0                     2.387309
1                         0.0                     0.000000

   authorities_contacted_Police  incident_state_NC  incident_state_NY  \
0                           0.0            3.05345                0.0
1                           0.0            0.00000                0.0

   incident_state_OH  incident_state_PA  incident_state_SC  incident_state_VA  \
0                0.0                0.0                0.0           0.000000
1                0.0                0.0                0.0           3.198324

   incident_state_WV  incident_city_Arlington  incident_city_Columbus  \
0                0.0                      0.0                     0.0
1                0.0                      0.0                     0.0

   incident_city_Hillsdale  incident_city_Northbend  incident_city_Northbrook  \
0                      0.0                 0.000000                       0.0
1                      0.0                 2.902839                       0.0

   incident_city_Riverwood  incident_city_Springfield  auto_make_Accura  \
0                 2.992602                        0.0               0.0
```

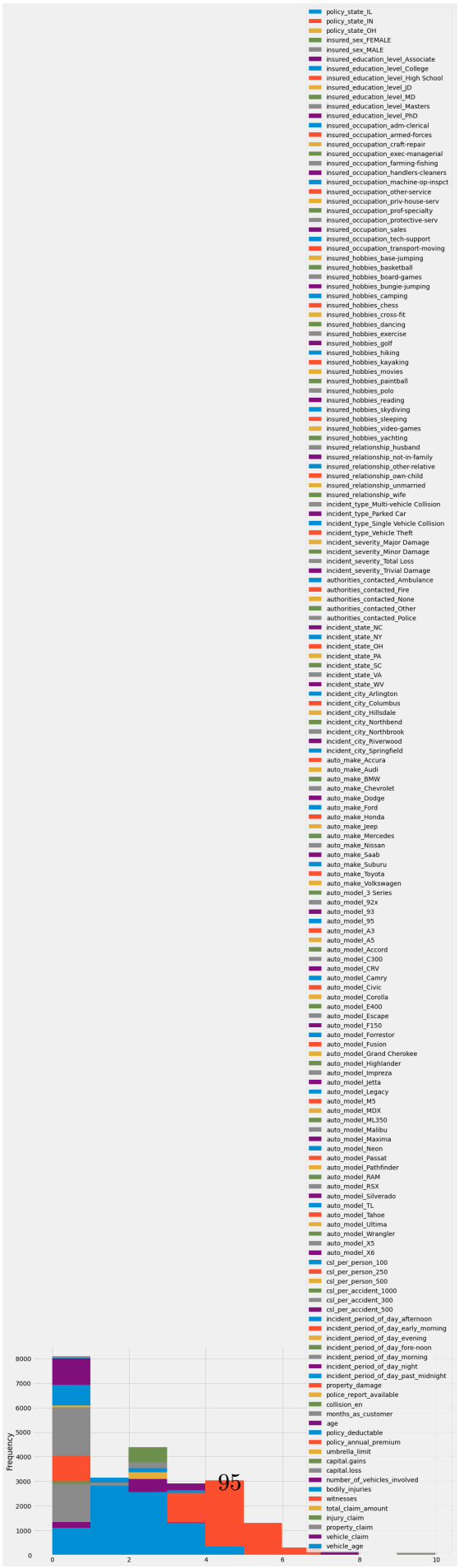
```
1                0.000000                         0.0               0.0

   auto_make_Audi  auto_make_BMW  auto_make_Chevrolet  auto_make_Dodge  \
0             0.0            0.0                  0.0              0.0
1             0.0            0.0                  0.0              0.0

   auto_make_Ford  auto_make_Honda  auto_make_Jeep  auto_make_Mercedes  \
0             0.0              0.0        4.148293                 0.0
1             0.0              0.0        0.000000                 0.0

   auto_make_Nissan  auto_make_Saab  auto_make_Suburu  auto_make_Toyota  \
0               0.0         0.00000               0.0               0.0
1               0.0         3.80693               0.0               0.0

   auto_make_Volkswagen  auto_model_3 Series  auto_model_92x  auto_model_93  \
0                   0.0                  0.0             0.0       0.000000
1                   0.0                  0.0             0.0       6.703527

   auto_model_95  auto_model_A3  auto_model_A5  auto_model_Accord  \
0            0.0            0.0            0.0                0.0
1            0.0            0.0            0.0                0.0

   auto_model_C300  auto_model_CRV  auto_model_Camry  auto_model_Civic  \
0              0.0             0.0               0.0               0.0
1              0.0             0.0               0.0               0.0

   auto_model_Corolla  auto_model_E400  auto_model_Escape  auto_model_F150  \
0                 0.0             0.0                0.0              0.0
1                 0.0             0.0                0.0              0.0

   auto_model_Forrestor  auto_model_Fusion  auto_model_Grand Cherokee  \
0                   0.0                0.0                        0.0
1                   0.0                0.0                        0.0

   auto_model_Highlander  auto_model_Impreza  auto_model_Jetta  \
0                    0.0                0.0               0.0
1                    0.0                0.0               0.0

   auto_model_Legacy  auto_model_M5  auto_model_MDX  auto_model_ML350  \
0                0.0            0.0             0.0               0.0
1                0.0            0.0             0.0               0.0

   auto_model_Malibu  auto_model_Maxima  auto_model_Neon  auto_model_Passat  \
0                0.0                0.0              0.0                0.0
1                0.0                0.0              0.0                0.0

   auto_model_Pathfinder  auto_model_RAM  auto_model_RSX  \
```

```
0                    0.0             0.0             0.0
1                    0.0             0.0             0.0

   auto_model_Silverado  auto_model_TL  auto_model_Tahoe  auto_model_Ultima  \
0                   0.0            0.0               0.0                0.0
1                   0.0            0.0               0.0                0.0

   auto_model_Wrangler  auto_model_X5  auto_model_X6  csl_per_person_100  \
0             5.440195            0.0            0.0                 0.0
1             0.000000            0.0            0.0                 0.0

   csl_per_person_250  csl_per_person_500  csl_per_accident_1000  \
0            2.088297                 0.0                    0.0
1            2.088297                 0.0                    0.0

   csl_per_accident_300  csl_per_accident_500  \
0                   0.0              2.088297
1                   0.0              2.088297

   incident_period_of_day_afternoon  incident_period_of_day_early_morning  \
0                           0.00000                                   0.0
1                           2.28319                                   0.0

   incident_period_of_day_evening  incident_period_of_day_fore-noon  \
0                             0.0                           2.87829
1                             0.0                           0.00000

   incident_period_of_day_morning  incident_period_of_day_night  \
0                             0.0                           0.0
1                             0.0                           0.0

   incident_period_of_day_past_midnight  property_damage  \
0                                   0.0         2.162213
1                                   0.0         0.000000

   police_report_available  collision_en  months_as_customer       age  \
0                      0.0      2.944422            0.338492  1.916405
1                      0.0      1.962948            0.443800  2.961716

   policy_deductable  policy_annual_premium  umbrella_limit  capital.gains  \
0           1.614670               4.316236        1.666924       0.000000
1           0.807335               4.126266        0.000000       0.003044

   capital.loss  number_of_vehicles_involved  bodily_injuries  witnesses  \
0      0.000000                     3.615905         3.342984   2.511628
1      1.385288                     0.903976         3.342984   2.511628
```

```
   total_claim_amount  injury_claim  property_claim  vehicle_claim  \
0            3.511278      0.701859        2.522132       3.580737
1            1.704751      2.698341        1.106835       2.793616

   vehicle_age
0     2.324003
1     1.394402
```

[ ]:
```
# Generate a Histogram plot on scaled data to check anomalies
plt.rcParams['figure.figsize'] = [15, 8]
X_train_scaled.plot(kind='hist')
```

[ ]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f45fe57d3c8>
```


policy_state_IL
policy_state_IN
policy_state_OH
insured_sex_FEMALE
insured_sex_MALE
insured_education_level_Associate
insured_education_level_College
insured_education_level_High School
insured_education_level_JD
insured_education_level_MD
insured_education_level_Masters
insured_education_level_PhD
insured_occupation_adm-clerical
insured_occupation_armed-forces
insured_occupation_craft-repair
insured_occupation_exec-managerial
insured_occupation_farming-fishing
insured_occupation_handlers-cleaners
insured_occupation_machine-op-inspct
insured_occupation_other-service
insured_occupation_priv-house-serv
insured_occupation_prof-specialty
insured_occupation_protective-serv
insured_occupation_sales
insured_occupation_tech-support
insured_occupation_transport-moving
insured_hobbies_base-jumping
insured_hobbies_basketball
insured_hobbies_board-games
insured_hobbies_bungie-jumping
insured_hobbies_camping
insured_hobbies_chess
insured_hobbies_cross-fit
insured_hobbies_dancing
insured_hobbies_exercise
insured_hobbies_golf
insured_hobbies_hiking
insured_hobbies_kayaking
insured_hobbies_movies
insured_hobbies_paintball
insured_hobbies_polo
insured_hobbies_reading
insured_hobbies_skydiving
insured_hobbies_sleeping
insured_hobbies_video-games
insured_hobbies_yachting
insured_relationship_husband
insured_relationship_not-in-family
insured_relationship_other-relative
insured_relationship_own-child
insured_relationship_unmarried
insured_relationship_wife
incident_type_Multi-vehicle Collision
incident_type_Parked Car
incident_type_Single Vehicle Collision
incident_type_Vehicle Theft
incident_severity_Major Damage
incident_severity_Minor Damage
incident_severity_Total Loss
incident_severity_Trivial Damage
authorities_contacted_Ambulance
authorities_contacted_Fire
authorities_contacted_None
authorities_contacted_Other
authorities_contacted_Police
incident_state_NC
incident_state_NY
incident_state_OH
incident_state_PA
incident_state_SC
incident_state_VA
incident_state_WV
incident_city_Arlington
incident_city_Columbus
incident_city_Hillsdale
incident_city_Northbend
incident_city_Northbrook
incident_city_Riverwood
incident_city_Springfield
auto_make_Accura
auto_make_Audi
auto_make_BMW
auto_make_Chevrolet
auto_make_Dodge
auto_make_Ford
auto_make_Honda
auto_make_Jeep
auto_make_Mercedes
auto_make_Nissan
auto_make_Saab
auto_make_Suburu
auto_make_Toyota
auto_make_Volkswagen
auto_model_3 Series
auto_model_92x
auto_model_93
auto_model_95
auto_model_A3
auto_model_A5
auto_model_Accord
auto_model_C300
auto_model_CRV
auto_model_Camry
auto_model_Civic
auto_model_Corolla
auto_model_E400
auto_model_Escape
auto_model_F150
auto_model_Forrestor
auto_model_Fusion
auto_model_Grand Cherokee
auto_model_Highlander
auto_model_Impreza
auto_model_Jetta
auto_model_Legacy
auto_model_M5
auto_model_MDX
auto_model_ML350
auto_model_Malibu
auto_model_Maxima
auto_model_Neon
auto_model_Passat
auto_model_Pathfinder
auto_model_RAM
auto_model_RSX
auto_model_Silverado
auto_model_TL
auto_model_Tahoe
auto_model_Ultima
auto_model_Wrangler
auto_model_X5
auto_model_X6
csl_per_person_100
csl_per_person_250
csl_per_person_500
csl_per_accident_1000
csl_per_accident_300
csl_per_accident_500
incident_period_of_day_afternoon
incident_period_of_day_early_morning
incident_period_of_day_evening
incident_period_of_day_fore-noon
incident_period_of_day_morning
incident_period_of_day_night
incident_period_of_day_past_midnight
property_damage
police_report_available
collision_en
months_as_customer
age
policy_deductable
policy_annual_premium
umbrella_limit
capital.gains
capital.loss
number_of_vehicles_involved
bodily_injuries
witnesses
total_claim_amount
injury_claim
property_claim
vehicle_claim
vehicle_age
Frequency
8000
7000
6000
5000
4000
3000
2000
1000
0
0
2
4
6
8
10
95

[ ]:
```
x_train_scaled = pd.DataFrame.to_numpy(X_train_scaled) # converting to array␣
↪for computational ease
x_train_scaled
```

[ ]:
```
array([[0.        , 0.        , 2.09909411, …, 2.52213226, 3.58073673,
        2.32400293],
       [0.        , 2.13021908, 0.        , …, 1.10683502, 2.79361616,
        1.39440176],
       [2.13684297, 0.        , 0.        , …, 2.26810455, 3.01916235,
        2.63386999],
       …,
       [0.        , 0.        , 2.09909411, …, 2.09391412, 3.31955118,
        1.85920235],
       [0.        , 0.        , 2.09909411, …, 2.1410907 , 2.267172  ,
        1.85920235],
       [2.13684297, 0.        , 0.        , …, 0.        , 0.98975575,
        2.47893646]])
```

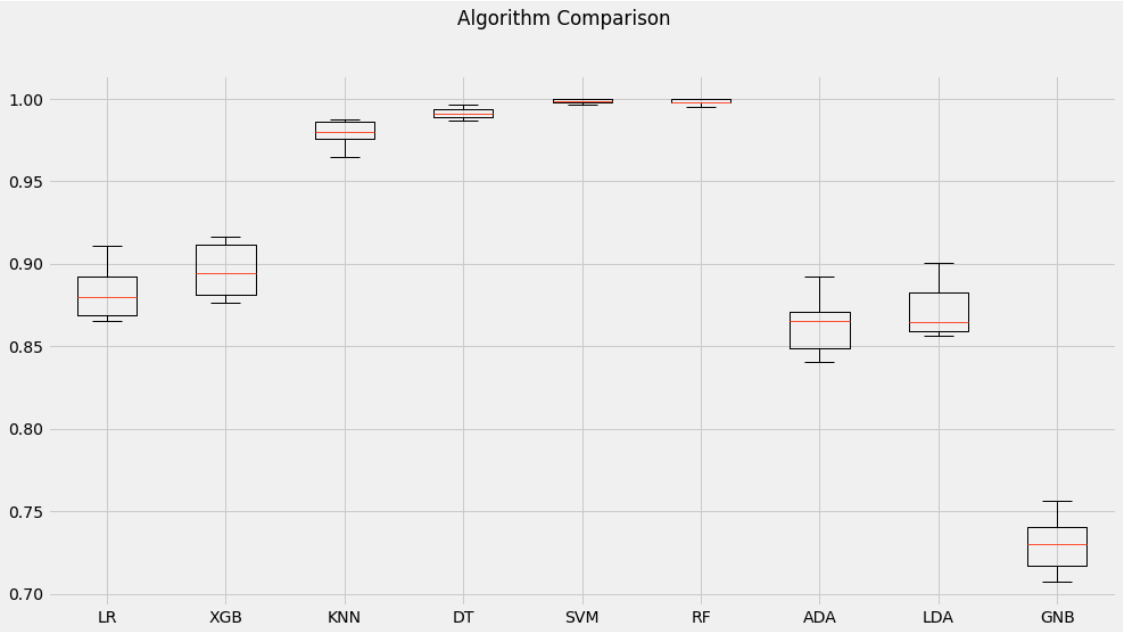
[ ]:
```
from sklearn.ensemble import AdaBoostClassifier, VotingClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn import model_selection
from sklearn.model_selection import KFold, cross_val_score
from xgboost import XGBClassifier
from sklearn.linear_model import LogisticRegressionCV
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC

xgb = XGBClassifier()
logreg= LogisticRegressionCV(solver='lbfgs', cv=10)
knn = KNeighborsClassifier(5)
svcl = SVC()
adb = AdaBoostClassifier()
dt = DecisionTreeClassifier(max_depth=5)
rf = RandomForestClassifier()
lda = LinearDiscriminantAnalysis()
gnb = GaussianNB()

# prepare configuration for cross validation test harness
seed = 7
# prepare models
models = []
models.append(('LR', LogisticRegressionCV(solver='lbfgs', max_iter=5000,␣
↪cv=10)))
```

```
models.append(('XGB', XGBClassifier()))
models.append(('KNN', KNeighborsClassifier()))
models.append(('DT', DecisionTreeClassifier()))
models.append(('SVM', SVC(gamma='auto')))
models.append(('RF', RandomForestClassifier(n_estimators=200)))
models.append(('ADA', AdaBoostClassifier(n_estimators=200)))
models.append(('LDA', LinearDiscriminantAnalysis()))
models.append(('GNB', GaussianNB()))

# evaluate each model in turn
results = []
names = []
scoring = 'accuracy'
for name, model in models:
    kfold = model_selection.KFold(n_splits=10, random_state=seed)
    cv_results = model_selection.cross_val_score(model, x_train_scaled,␣
 ↪y_train, cv=kfold, scoring=scoring)
    results.append(cv_results)
    names.append(name)
    msg = "%s: %f (%f)" % (name, cv_results.mean(), cv_results.std())
    print(msg)

# boxplot algorithm comparison
plt.rcParams['figure.figsize'] = [15, 8]
fig = plt.figure()
fig.suptitle('Algorithm Comparison')
ax = fig.add_subplot(111)
plt.boxplot(results)
ax.set_xticklabels(names)
plt.show()
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


LR: 0.882346 (0.015079)

/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.
```

```
XGB: 0.895814 (0.015307)
KNN: 0.979431 (0.007119)

/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


DT: 0.991063 (0.003333)

/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


SVM: 0.998654 (0.001390)

/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


RF: 0.998286 (0.001469)

/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


ADA: 0.862634 (0.015827)

/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


LDA: 0.872308 (0.016027)
GNB: 0.731145 (0.015877)
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.
```



[ ]:
```
 clf1= LogisticRegressionCV(solver='lbfgs', max_iter=5000, cv=10)
clf2 = XGBClassifier()

clf = [
    ('LR', clf1),
    ('XGB', clf2)]

#create our voting classifier, inputting our models
eclf= VotingClassifier(estimators=[
    ('LR', clf1),
    ('XGB', clf2)], voting='hard')

for clf, label in zip([clf1, clf2, eclf], [
    'Logistic Regression',
    'XGB Classifier',
    'Ensemble']):

    scores = cross_val_score(clf, x_train_scaled, y_train, cv=10,␣
 ↪scoring='accuracy')
```

```
    print("Accuracy: %0.2f (+/- %0.2f) [%s]" % (scores.mean(), scores.std(),␣
↪label))
```

```
Accuracy: 0.88 (+/- 0.01) [Logistic Regression]
Accuracy: 0.90 (+/- 0.02) [XGB Classifier]
Accuracy: 0.89 (+/- 0.02) [Ensemble]
```

[ ]:
```
from numpy import sort
from sklearn.feature_selection import SelectFromModel

# fit model on all training data
xgb = XGBClassifier()
xgb.fit(x_train_scaled, y_train)

# make predictions for test data and evaluate
xgb_pred = xgb.predict(X_test_scaled)
predictions = [round(value) for value in xgb_pred]
accuracy = accuracy_score(y_test, xgb_pred)
print("Accuracy: %.2f%%" % (accuracy * 100.0))

# Fit model using each importance as a threshold
thresholds = sort(xgb.feature_importances_)
for thresh in thresholds:

    # select features using threshold
    selection = SelectFromModel(xgb, threshold=thresh, prefit=True)
    select_X_train = selection.transform(x_train_scaled)

    # train model
    selection_model = XGBClassifier()
    selection_model.fit(select_X_train, y_train)

    # eval model
    select_X_test = selection.transform(X_test_scaled)
    xgb_pred = selection_model.predict(select_X_test)
    predictions = [round(value) for value in xgb_pred]
    accuracy = accuracy_score(y_test, xgb_pred)
    print("Thresh=%.3f, n=%d, Accuracy: %.2f%%" % (thresh, select_X_train.
↪shape[1], accuracy*100.0))
```

```
Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
```

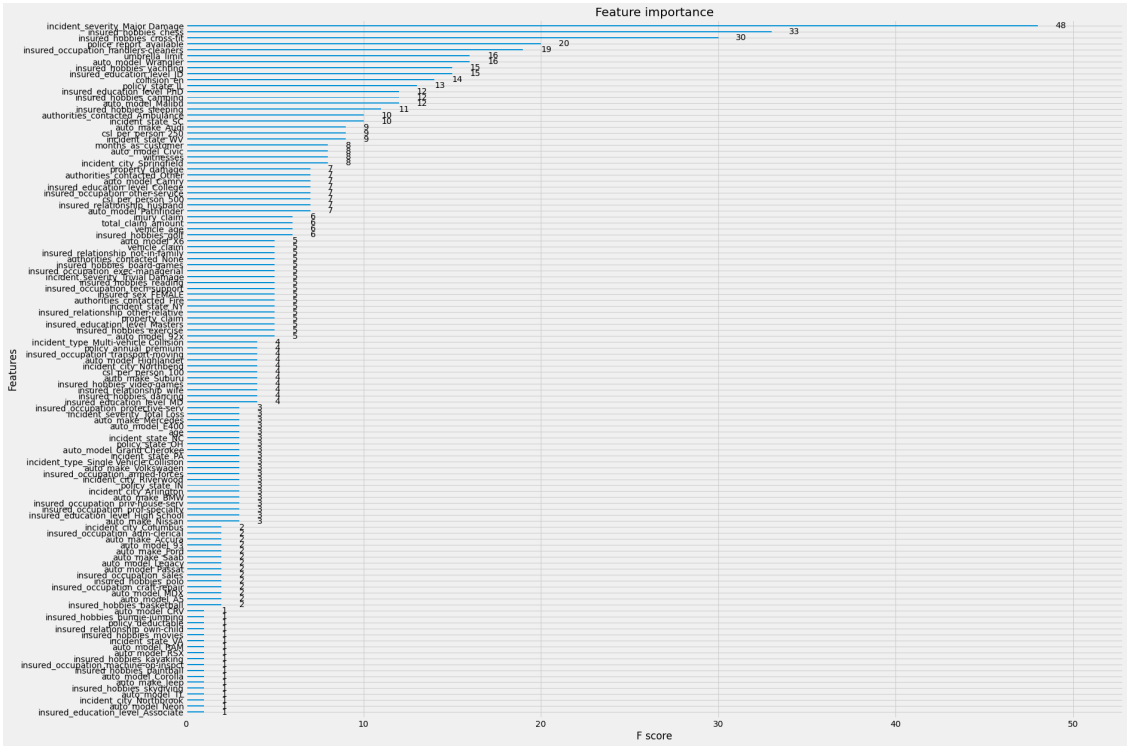
```
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.000, n=163, Accuracy: 88.55%
Thresh=0.001, n=116, Accuracy: 88.55%
Thresh=0.001, n=115, Accuracy: 88.64%
Thresh=0.002, n=114, Accuracy: 88.55%
Thresh=0.003, n=113, Accuracy: 88.35%
Thresh=0.003, n=112, Accuracy: 88.50%
Thresh=0.003, n=111, Accuracy: 88.25%
Thresh=0.003, n=110, Accuracy: 88.60%
Thresh=0.003, n=109, Accuracy: 88.45%
```

```
Thresh=0.003, n=108, Accuracy: 88.55%
Thresh=0.004, n=107, Accuracy: 88.15%
Thresh=0.004, n=106, Accuracy: 88.20%
Thresh=0.004, n=105, Accuracy: 88.40%
Thresh=0.004, n=104, Accuracy: 88.40%
Thresh=0.004, n=103, Accuracy: 87.86%
Thresh=0.004, n=102, Accuracy: 88.79%
Thresh=0.004, n=101, Accuracy: 88.55%
Thresh=0.004, n=100, Accuracy: 88.69%
Thresh=0.004, n=99, Accuracy: 89.23%
Thresh=0.004, n=98, Accuracy: 88.79%
Thresh=0.004, n=97, Accuracy: 88.25%
Thresh=0.004, n=96, Accuracy: 88.84%
Thresh=0.004, n=95, Accuracy: 88.50%
Thresh=0.004, n=94, Accuracy: 88.40%
Thresh=0.004, n=93, Accuracy: 88.94%
Thresh=0.004, n=92, Accuracy: 88.74%
Thresh=0.004, n=91, Accuracy: 88.69%
Thresh=0.004, n=90, Accuracy: 88.20%
Thresh=0.004, n=89, Accuracy: 88.30%
Thresh=0.004, n=88, Accuracy: 88.64%
Thresh=0.004, n=87, Accuracy: 88.64%
Thresh=0.005, n=86, Accuracy: 88.69%
Thresh=0.005, n=85, Accuracy: 88.74%
Thresh=0.005, n=84, Accuracy: 88.99%
Thresh=0.005, n=83, Accuracy: 88.55%
Thresh=0.005, n=82, Accuracy: 88.55%
Thresh=0.005, n=81, Accuracy: 88.45%
Thresh=0.005, n=80, Accuracy: 88.69%
Thresh=0.005, n=79, Accuracy: 88.89%
Thresh=0.005, n=78, Accuracy: 88.35%
Thresh=0.005, n=77, Accuracy: 88.25%
Thresh=0.005, n=76, Accuracy: 88.35%
Thresh=0.005, n=75, Accuracy: 88.20%
Thresh=0.005, n=74, Accuracy: 87.86%
Thresh=0.005, n=73, Accuracy: 88.11%
Thresh=0.005, n=72, Accuracy: 88.06%
Thresh=0.005, n=71, Accuracy: 87.81%
Thresh=0.005, n=70, Accuracy: 88.50%
Thresh=0.006, n=69, Accuracy: 88.25%
Thresh=0.006, n=68, Accuracy: 88.01%
Thresh=0.006, n=67, Accuracy: 88.11%
Thresh=0.006, n=66, Accuracy: 87.86%
Thresh=0.006, n=65, Accuracy: 87.71%
Thresh=0.006, n=64, Accuracy: 88.40%
Thresh=0.006, n=63, Accuracy: 87.96%
Thresh=0.006, n=62, Accuracy: 88.25%
Thresh=0.006, n=61, Accuracy: 87.71%
```

```
Thresh=0.006, n=60, Accuracy: 88.40%
Thresh=0.006, n=59, Accuracy: 87.67%
Thresh=0.006, n=58, Accuracy: 87.57%
Thresh=0.006, n=57, Accuracy: 87.57%
Thresh=0.006, n=56, Accuracy: 87.67%
Thresh=0.006, n=55, Accuracy: 87.62%
Thresh=0.006, n=54, Accuracy: 87.91%
Thresh=0.006, n=53, Accuracy: 87.86%
Thresh=0.006, n=52, Accuracy: 87.62%
Thresh=0.007, n=51, Accuracy: 87.57%
Thresh=0.007, n=50, Accuracy: 87.37%
Thresh=0.007, n=49, Accuracy: 87.62%
Thresh=0.007, n=48, Accuracy: 87.13%
Thresh=0.007, n=47, Accuracy: 87.18%
Thresh=0.007, n=46, Accuracy: 86.98%
Thresh=0.007, n=45, Accuracy: 86.98%
Thresh=0.007, n=44, Accuracy: 86.98%
Thresh=0.007, n=43, Accuracy: 86.98%
Thresh=0.007, n=42, Accuracy: 87.18%
Thresh=0.008, n=41, Accuracy: 86.98%
Thresh=0.008, n=40, Accuracy: 87.81%
Thresh=0.008, n=39, Accuracy: 87.57%
Thresh=0.008, n=38, Accuracy: 87.47%
Thresh=0.008, n=37, Accuracy: 87.27%
Thresh=0.008, n=36, Accuracy: 86.83%
Thresh=0.008, n=35, Accuracy: 87.08%
Thresh=0.008, n=34, Accuracy: 87.08%
Thresh=0.008, n=33, Accuracy: 87.27%
Thresh=0.008, n=32, Accuracy: 87.27%
Thresh=0.008, n=31, Accuracy: 86.69%
Thresh=0.009, n=30, Accuracy: 86.88%
Thresh=0.009, n=29, Accuracy: 86.69%
Thresh=0.009, n=28, Accuracy: 86.83%
Thresh=0.009, n=27, Accuracy: 86.78%
Thresh=0.009, n=26, Accuracy: 86.59%
Thresh=0.009, n=25, Accuracy: 86.69%
Thresh=0.009, n=24, Accuracy: 87.03%
Thresh=0.009, n=23, Accuracy: 87.13%
Thresh=0.010, n=22, Accuracy: 86.25%
Thresh=0.010, n=21, Accuracy: 86.39%
Thresh=0.010, n=20, Accuracy: 86.34%
Thresh=0.011, n=19, Accuracy: 86.34%
Thresh=0.011, n=18, Accuracy: 86.34%
Thresh=0.012, n=17, Accuracy: 86.39%
Thresh=0.012, n=16, Accuracy: 85.95%
Thresh=0.012, n=15, Accuracy: 86.10%
Thresh=0.013, n=14, Accuracy: 86.10%
Thresh=0.013, n=13, Accuracy: 86.05%
```

```
Thresh=0.014, n=12, Accuracy: 86.00%
Thresh=0.016, n=11, Accuracy: 85.81%
Thresh=0.016, n=10, Accuracy: 85.76%
Thresh=0.016, n=9, Accuracy: 85.66%
Thresh=0.017, n=8, Accuracy: 85.61%
Thresh=0.017, n=7, Accuracy: 85.66%
Thresh=0.017, n=6, Accuracy: 85.46%
Thresh=0.018, n=5, Accuracy: 85.61%
Thresh=0.024, n=4, Accuracy: 85.36%
Thresh=0.055, n=3, Accuracy: 85.36%
Thresh=0.058, n=2, Accuracy: 80.52%
Thresh=0.089, n=1, Accuracy: 74.60%
```

[ ]:
```
from xgboost import plot_importance
x = XGBClassifier()
x.fit(X_train_scaled, y_train) # fitting the model again on dataframe to identify the feature names

plt.rcParams['figure.figsize'] = [25, 20]
# plot feature importance
plot_importance(x);
```



[ ]:
```
from pprint import pprint
# Check parameters used
print('Parameters currently in use:\n')
pprint(x.get_params())
```

```
Parameters currently in use:

{'base_score': 0.5,
 'booster': 'gbtree',
 'colsample_bylevel': 1,
 'colsample_bynode': 1,
 'colsample_bytree': 1,
 'gamma': 0,
 'learning_rate': 0.1,
 'max_delta_step': 0,
 'max_depth': 3,
 'min_child_weight': 1,
 'missing': None,
 'n_estimators': 100,
 'n_jobs': 1,
 'nthread': None,
 'objective': 'binary:logistic',
 'random_state': 0,
 'reg_alpha': 0,
 'reg_lambda': 1,
 'scale_pos_weight': 1,
 'seed': None,
 'silent': None,
 'subsample': 1,
 'verbosity': 1}
```

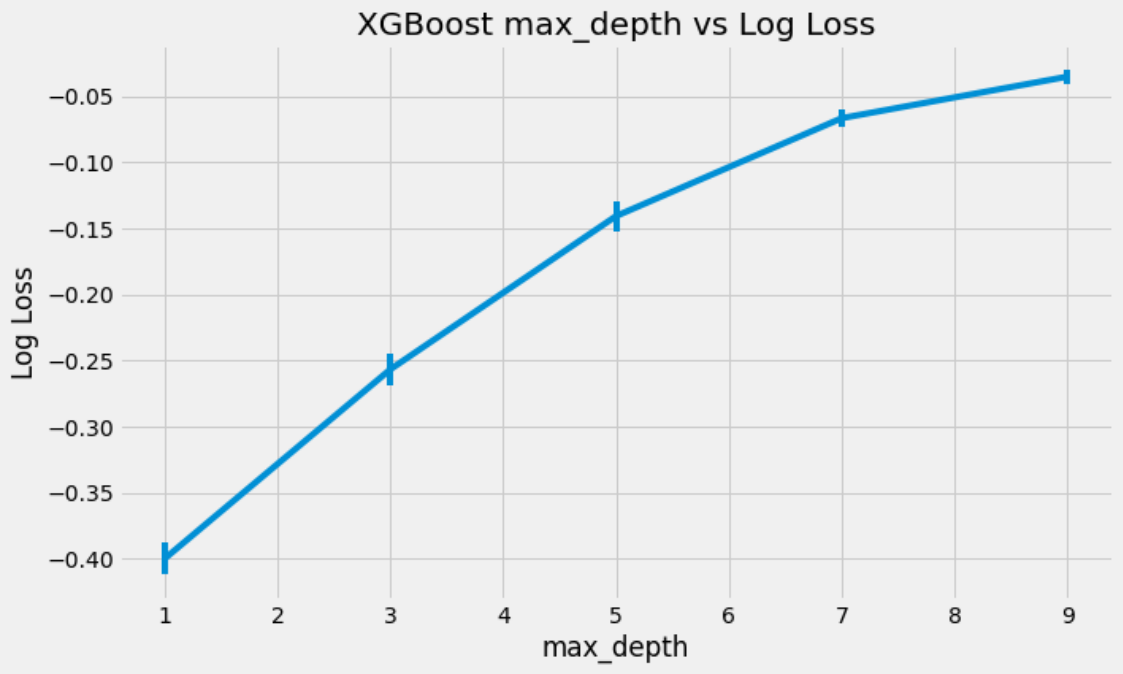
[ ]:
```
from sklearn.model_selection import StratifiedKFold
from sklearn.model_selection import GridSearchCV
import matplotlib
matplotlib.use('Agg')
from matplotlib import pyplot
plt.rcParams['figure.figsize'] = [10, 6]

# grid search
max_depth = range(1, 11, 2)
print(max_depth)

param_grid = dict(max_depth=max_depth)
kfold = StratifiedKFold(n_splits=10, shuffle=True, random_state=7)
grid_search = GridSearchCV(xgb, param_grid, scoring="neg_log_loss", n_jobs=-1,␣
 ↪cv=kfold, verbose=1,  iid=False)
grid_result = grid_search.fit(x_train_scaled, y_train)
```

```
# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))


# plot
pyplot.errorbar(max_depth, means, yerr=stds)
pyplot.title("XGBoost max_depth vs Log Loss")
pyplot.xlabel('max_depth')
pyplot.ylabel('Log Loss')
```

```
range(1, 11, 2)
Fitting 10 folds for each of 5 candidates, totalling 50 fits
```

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 2 concurrent workers.
[Parallel(n_jobs=-1)]: Done  46 tasks      | elapsed:  2.0min
[Parallel(n_jobs=-1)]: Done  50 out of  50 | elapsed:  2.3min finished
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_search.py:823:
FutureWarning:

The parameter 'iid' is deprecated in 0.22 and will be removed in 0.24.


Best: -0.035376 using {'max_depth': 9}
-0.399755 (0.011874) with: {'max_depth': 1}
-0.256650 (0.011923) with: {'max_depth': 3}
-0.140782 (0.011174) with: {'max_depth': 5}
-0.066761 (0.007062) with: {'max_depth': 7}
-0.035376 (0.005156) with: {'max_depth': 9}
```

```
[ ]: Text(0, 0.5, 'Log Loss')
```

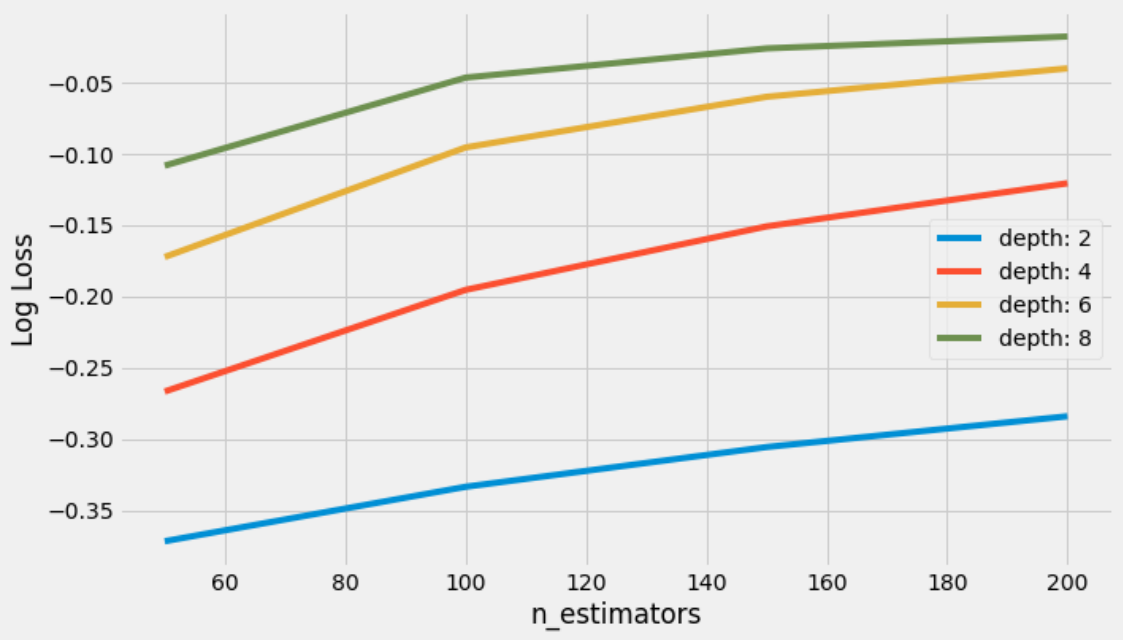


[ ]:
```
import numpy

n_estimators = [50, 100, 150, 200]
max_depth = [2, 4, 6, 8]
print(max_depth)
param_grid = dict(max_depth=max_depth, n_estimators=n_estimators)
kfold = StratifiedKFold(n_splits=10, shuffle=True, random_state=7)
grid_search = GridSearchCV(xgb, param_grid, scoring="neg_log_loss", n_jobs=-1,␣
 ↪cv=kfold, verbose=1, iid=False)
grid_result = grid_search.fit(x_train_scaled, y_train)

# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))

# plot results
scores = numpy.array(means).reshape(len(max_depth), len(n_estimators))
for i, value in enumerate(max_depth):
    pyplot.plot(n_estimators, scores[i], label='depth: ' + str(value))
pyplot.legend()
pyplot.xlabel('n_estimators')
```

```
pyplot.ylabel('Log Loss')
```

```
[2, 4, 6, 8]
Fitting 10 folds for each of 16 candidates, totalling 160 fits

[Parallel(n_jobs=-1)]: Using backend LokyBackend with 2 concurrent workers.
[Parallel(n_jobs=-1)]: Done  46 tasks      | elapsed:  1.1min
[Parallel(n_jobs=-1)]: Done 160 out of 160 | elapsed:  9.0min finished
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_search.py:823:
FutureWarning:

The parameter 'iid' is deprecated in 0.22 and will be removed in 0.24.


Best: -0.017716 using {'max_depth': 8, 'n_estimators': 200}
-0.371660 (0.011922) with: {'max_depth': 2, 'n_estimators': 50}
-0.333465 (0.012973) with: {'max_depth': 2, 'n_estimators': 100}
-0.305607 (0.012730) with: {'max_depth': 2, 'n_estimators': 150}
-0.284021 (0.013709) with: {'max_depth': 2, 'n_estimators': 200}
-0.266671 (0.013832) with: {'max_depth': 4, 'n_estimators': 50}
-0.195383 (0.013091) with: {'max_depth': 4, 'n_estimators': 100}
-0.150867 (0.011222) with: {'max_depth': 4, 'n_estimators': 150}
-0.120623 (0.010872) with: {'max_depth': 4, 'n_estimators': 200}
-0.172327 (0.010554) with: {'max_depth': 6, 'n_estimators': 50}
-0.095497 (0.008775) with: {'max_depth': 6, 'n_estimators': 100}
-0.059983 (0.006085) with: {'max_depth': 6, 'n_estimators': 150}
-0.040057 (0.004324) with: {'max_depth': 6, 'n_estimators': 200}
-0.108258 (0.007588) with: {'max_depth': 8, 'n_estimators': 50}
-0.046535 (0.003471) with: {'max_depth': 8, 'n_estimators': 100}
-0.026046 (0.002697) with: {'max_depth': 8, 'n_estimators': 150}
-0.017716 (0.002372) with: {'max_depth': 8, 'n_estimators': 200}
```

[ ]: Text(0, 0.5, 'Log Loss')



[ ]:
```
xgb = XGBClassifier(objective='binary:logistic', random_state=7, n_jobs=-1)
xgb.fit(x_train_scaled, y_train)
scores = cross_val_score(xgb, x_train_scaled, y_train, cv=kfold,␣
 ↪scoring='brier_score_loss')
print('Brier loss:', "{0:.5f}".format(np.mean(scores)*-1))
```

```
Brier loss: 0.07136
```

[ ]:
```
print(xgb.get_params())
```

```
{'base_score': 0.5, 'booster': 'gbtree', 'colsample_bylevel': 1,
'colsample_bynode': 1, 'colsample_bytree': 1, 'gamma': 0, 'learning_rate': 0.1,
'max_delta_step': 0, 'max_depth': 3, 'min_child_weight': 1, 'missing': None,
'n_estimators': 100, 'n_jobs': -1, 'nthread': None, 'objective':
'binary:logistic', 'random_state': 7, 'reg_alpha': 0, 'reg_lambda': 1,
'scale_pos_weight': 1, 'seed': None, 'silent': None, 'subsample': 1,
'verbosity': 1}
```

[ ]:
```
from sklearn.model_selection import RandomizedSearchCV

# Create the parameter grid
params = {
    'learning_rate': [0.0001, 0.001, 0.01, 0.1, 0.2, 0.3],
    'n_estimators': [int(x) for x in np.linspace(start=100, stop=1200, num=9)],
    'max_depth': [i for i in range(3, 10)],
    'min_child_weight': [i for i in range(1, 7)],
```

```
    'subsample': [i/10.0 for i in range(6,11)],
    'colsample_bytree': [i/10.0 for i in range(6,11)]
}

# Create the randomised grid search model
# "n_iter = number of parameter settings that are sampled. n_iter trades off␣
 ↪runtime vs quality of the solution"
rgs = RandomizedSearchCV(estimator=xgb, param_distributions=params, n_iter=20,␣
 ↪cv=kfold,
                         random_state=7, n_jobs=-1,
                         scoring='brier_score_loss', return_train_score=True)
# Fit rgs
rgs.fit(x_train_scaled, y_train)

# Print results
print(rgs)
```

```
RandomizedSearchCV(cv=StratifiedKFold(n_splits=10, random_state=7,
shuffle=True),
                   error_score=nan,
                   estimator=XGBClassifier(base_score=0.5, booster='gbtree',
                                           colsample_bylevel=1,
                                           colsample_bynode=1,
                                           colsample_bytree=1, gamma=0,
                                           learning_rate=0.1, max_delta_step=0,
                                           max_depth=3, min_child_weight=1,
                                           missing=None, n_estimators=100,
                                           n_jobs=-1, nthread=None,
                                           objective='bin…
                   param_distributions={'colsample_bytree': [0.6, 0.7, 0.8, 0.9,
                                                             1.0],
                                        'learning_rate': [0.0001, 0.001, 0.01,
                                                          0.1, 0.2, 0.3],
                                        'max_depth': [3, 4, 5, 6, 7, 8, 9],
                                        'min_child_weight': [1, 2, 3, 4, 5, 6],
                                        'n_estimators': [100, 237, 375, 512,
                                                         650, 787, 925, 1062,
                                                         1200],
                                        'subsample': [0.6, 0.7, 0.8, 0.9, 1.0]},
                   pre_dispatch='2*n_jobs', random_state=7, refit=True,
                   return_train_score=True, scoring='brier_score_loss',
                   verbose=0)
```

[ ]:
```
best_score = rgs.best_score_
best_params = rgs.best_params_
print("Best score: {}".format(best_score))
print("Best params: ")
```

```
for param_name in sorted(best_params.keys()):
    print('%s: %r' % (param_name, best_params[param_name]))
```

```
Best score: -0.004253326756930206
Best params:
colsample_bytree: 1.0
learning_rate: 0.2
max_depth: 5
min_child_weight: 3
n_estimators: 1200
subsample: 0.9
```

[ ]:
```
# make predictions for test data and evaluate
rgs_pred = rgs.predict(X_test_scaled)

print('Accuracy: ', round(accuracy_score(y_test, rgs_pred)*100, 2))
print( 'Cohen Kappa: '+ str(np.round(cohen_kappa_score(y_test, rgs_pred),3)))
print('Recall: ', round(recall_score(y_test, rgs_pred)*100, 2))
print('\n Classification Report:\n', classification_report(y_test, rgs_pred))

print(result.mean())
```

```
Accuracy:  99.76
Cohen Kappa: 0.995
Recall:  100.0

 Classification Report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00      1037
           1       1.00      1.00      1.00      1006

    accuracy                           1.00      2043
   macro avg       1.00      1.00      1.00      2043
weighted avg       1.00      1.00      1.00      2043

0.8486929162094305
```

[ ]:
```
xgb = XGBClassifier()

# prepare configuration for cross validation test harness
seed = 7
# prepare models
models = []
models.append(('XGB', XGBClassifier()))

# evaluate each model in turn
```

```
results = []
names = []
scoring = 'accuracy'
for name, model in models:
    kfold = model_selection.KFold(n_splits=10, random_state=seed)
    cv_results = model_selection.cross_val_score(model, x_train_scaled,␣
 ↪y_train, cv=kfold, scoring=scoring)
    results.append(cv_results)
    names.append(name)
    msg = "%s: %f (%f)" % (name, cv_results.mean(), cv_results.std())
    print(msg)
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


XGB: 0.895814 (0.015307)
```

[ ]:
```
# Fit rgs
model.fit(x_train_scaled, y_train)

# make predictions for test data
y_pred = model.predict(X_test_scaled)
predictions = [round(value) for value in y_pred]

# evaluate predictions
accuracy = accuracy_score(y_test, predictions)
print("Accuracy: %.2f%%" % (accuracy * 100.0))
```

```
Accuracy: 88.55%
```

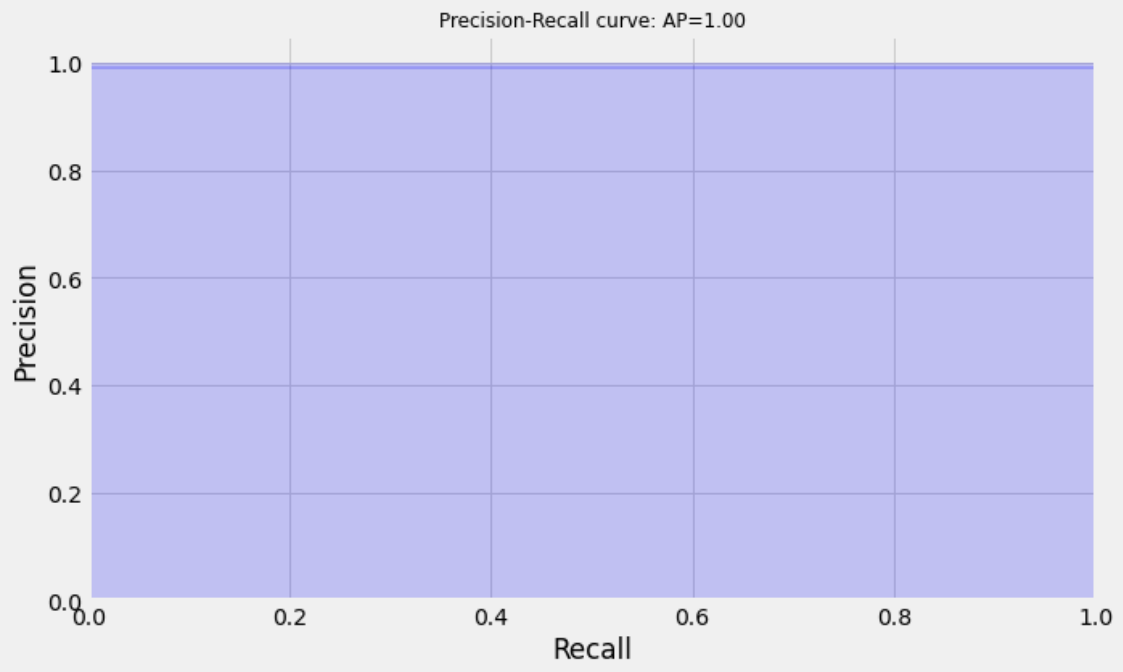
[ ]:
```
from sklearn.metrics import average_precision_score
average_precision = average_precision_score(y_test, rgs_pred)

print('Average precision-recall score: {0:0.2f}'.format(
      average_precision))
```

```
Average precision-recall score: 1.00
```

[ ]:
```
from sklearn.metrics import precision_recall_curve
from inspect import signature

plt.rcParams['figure.figsize'] = [10, 6]
```

```
precision, recall, _ = precision_recall_curve(y_test, rgs_pred)

step_kwargs = ({'step': 'post'}
               if 'step' in signature(plt.fill_between).parameters
               else {})
plt.step(recall, precision, color='b', alpha=0.2,where='post')
plt.fill_between(recall, precision, alpha=0.2, color='b', **step_kwargs)
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.ylim([0.0, 1.05])
plt.xlim([0.0, 1.0])
plt.title('Precision-Recall curve: AP={0:0.2f}'.format(average_precision),␣
 ↪fontsize=12)
```

[ ]:
```
Text(0.5, 1.0, 'Precision-Recall curve: AP=1.00')
```

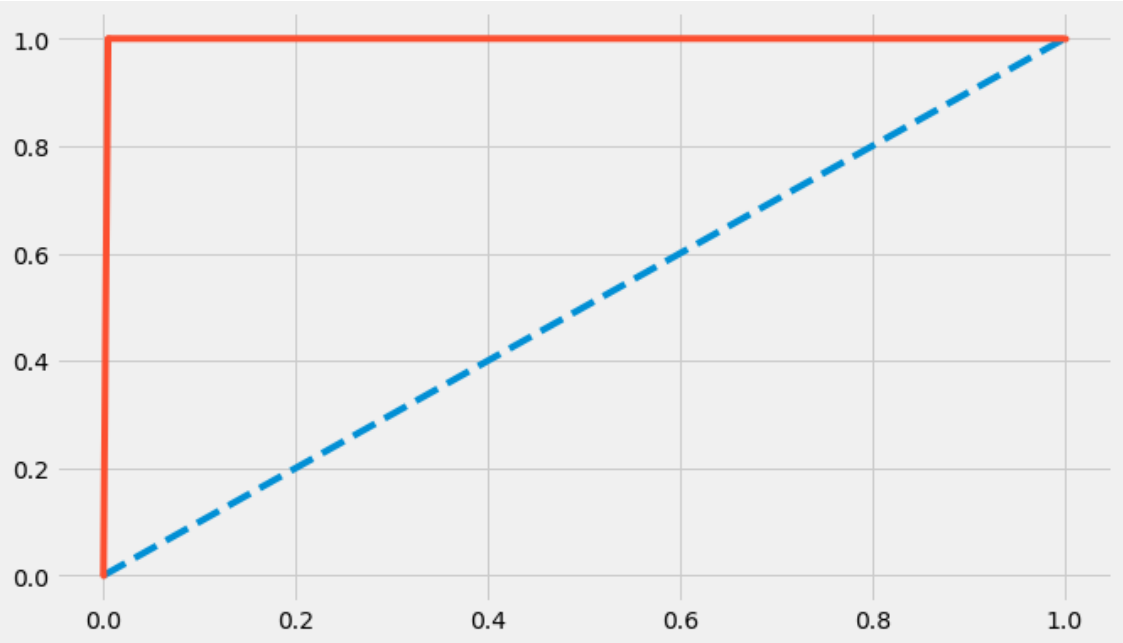


[ ]:
```
from sklearn.metrics import roc_curve
from sklearn.metrics import roc_auc_score

# calculate AUC
auc = roc_auc_score(y_test, rgs_pred)
print('AUC: %.3f' % auc)

# calculate roc curve
fpr, tpr, thresholds = roc_curve(y_test, rgs_pred)
```

```
# plot no skill
plt.rcParams['figure.figsize'] = [10, 6]
plt.plot([0, 1], [0, 1], linestyle='--')

# plot the roc curve for the model
plt.plot(fpr, tpr, marker='.')
```

```
AUC: 0.998
```

[ ]:
```
[<matplotlib.lines.Line2D at 0x7f45ea737208>]
```

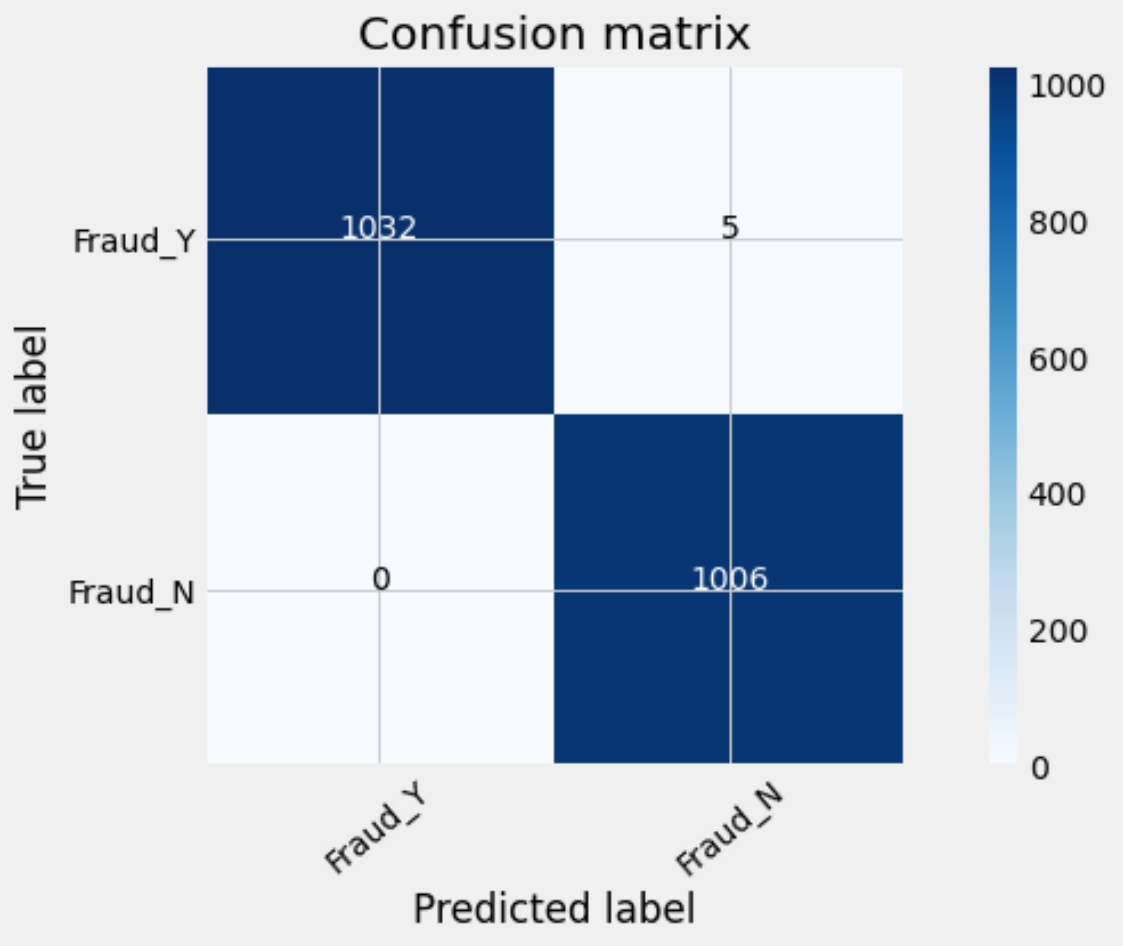


[ ]:
```
from sklearn.metrics import confusion_matrix
import itertools

#Evaluation of Model - Confusion Matrix Plot
def plot_confusion_matrix(cm, classes, title ='Confusion matrix', normalize =␣
 ↪False, cmap = plt.cm.Blues):

    print('Confusion matrix')

    print(cm)

    plt.style.use('fivethirtyeight')
    fig = plt.figure(figsize=(10,6))
```

```
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=40)
    plt.yticks(tick_marks, classes)

    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.ylabel('True label')
    plt.xlabel('Predicted label')
    plt.tight_layout()


# Compute confusion matrix
cnf_matrix = confusion_matrix(y_test, rgs_pred)
np.set_printoptions(precision=2)

# Plot confusion matrix
plt.figure()
plot_confusion_matrix(cnf_matrix, classes=['Fraud_Y','Fraud_N'],
                      title='Confusion matrix')
```

```
Confusion matrix
[[1032    5]
 [   0 1006]]
```

```
<Figure size 720x432 with 0 Axes>
```



[ ]:
```
from sklearn.feature_selection import VarianceThreshold

constant_filter = VarianceThreshold(threshold=0.057)
constant_filter.fit(X_train_scaled)

constant_columns = [column for column in X_train_scaled.columns
                    if column not in X_train_scaled.columns[constant_filter.
 ↪get_support()]]

print(len(constant_columns))
```

0

[ ]:
```
correlated_features = set()
correlation_matrix = X_train_scaled.corr()

for i in range(len(correlation_matrix .columns)):
    for j in range(i):
```

```
        if abs(correlation_matrix.iloc[i, j]) > 0.8:
            colname = correlation_matrix.columns[i]
            correlated_features.add(colname)

len(correlated_features)
```

[ ]: 4

[ ]:
```
print(correlated_features)
```

```
{'insured_sex_MALE', 'csl_per_accident_300', 'csl_per_accident_1000',
'csl_per_accident_500'}
```

[ ]:
```
X.head(1)
```

[ ]:
```
   policy_state_IL  policy_state_IN  policy_state_OH  insured_sex_FEMALE  \
0                0                1                0                   1

   insured_sex_MALE  insured_education_level_Associate  \
0                 0                                  1

   insured_education_level_College  insured_education_level_High School  \
0                                0                                     0

   insured_education_level_JD  insured_education_level_MD  \
0                           0                           0

   insured_education_level_Masters  insured_education_level_PhD  \
0                                0                           0

   insured_occupation_adm-clerical  insured_occupation_armed-forces  \
0                                0                                0

   insured_occupation_craft-repair  insured_occupation_exec-managerial  \
0                                0                                  0

   insured_occupation_farming-fishing  insured_occupation_handlers-cleaners  \
0                                   0                                     0

   insured_occupation_machine-op-inspct  insured_occupation_other-service  \
0                                     0                                  0

   insured_occupation_priv-house-serv  insured_occupation_prof-specialty  \
0                                   1                                  0

   insured_occupation_protective-serv  insured_occupation_sales  \
0                                   0                        0
```

```
   insured_occupation_tech-support  insured_occupation_transport-moving  \
0                                0                                    0

   insured_hobbies_base-jumping  insured_hobbies_basketball  \
0                             0                           0

   insured_hobbies_board-games  insured_hobbies_bungie-jumping  \
0                            0                               0

   insured_hobbies_camping  insured_hobbies_chess  insured_hobbies_cross-fit  \
0                        0                      0                          0

   insured_hobbies_dancing  insured_hobbies_exercise  insured_hobbies_golf  \
0                        0                         0                     0

   insured_hobbies_hiking  insured_hobbies_kayaking  insured_hobbies_movies  \
0                       0                         0                       1

   insured_hobbies_paintball  insured_hobbies_polo  insured_hobbies_reading  \
0                          0                     0                        0

   insured_hobbies_skydiving  insured_hobbies_sleeping  \
0                          0                        0

   insured_hobbies_video-games  insured_hobbies_yachting  \
0                            0                       0

   insured_relationship_husband  insured_relationship_not-in-family  \
0                             1                                    0

   insured_relationship_other-relative  insured_relationship_own-child  \
0                                    0                                0

   insured_relationship_unmarried  insured_relationship_wife  \
0                               0                          0

   incident_type_Multi-vehicle Collision  incident_type_Parked Car  \
0                                      0                         0

   incident_type_Single Vehicle Collision  incident_type_Vehicle Theft  \
0                                       1                            0

   incident_severity_Major Damage  incident_severity_Minor Damage  \
0                               0                               1

   incident_severity_Total Loss  incident_severity_Trivial Damage  \
```

```
0                             0                                   0

   authorities_contacted_Ambulance  authorities_contacted_Fire  \
0                               0                           0

   authorities_contacted_None  authorities_contacted_Other  \
0                           0                           1

   authorities_contacted_Police  incident_state_NC  incident_state_NY  \
0                             0                 0                  1

   incident_state_OH  incident_state_PA  incident_state_SC  incident_state_VA  \
0                  0                  0                  0                  0

   incident_state_WV  incident_city_Arlington  incident_city_Columbus  \
0                  0                        0                       0

   incident_city_Hillsdale  incident_city_Northbend  incident_city_Northbrook  \
0                        1                        0                         0

   incident_city_Riverwood  incident_city_Springfield  auto_make_Accura  \
0                        0                          0                 0

   auto_make_Audi  auto_make_BMW  auto_make_Chevrolet  auto_make_Dodge  \
0               0              0                    0                0

   auto_make_Ford  auto_make_Honda  auto_make_Jeep  auto_make_Mercedes  \
0               0                0               0                   0

   auto_make_Nissan  auto_make_Saab  auto_make_Suburu  auto_make_Toyota  \
0                 0               1                 0                 0

   auto_make_Volkswagen  auto_model_3 Series  auto_model_92x  auto_model_93  \
0                     0                    0               1              0

   auto_model_95  auto_model_A3  auto_model_A5  auto_model_Accord  \
0              0              0              0                  0

   auto_model_C300  auto_model_CRV  auto_model_Camry  auto_model_Civic  \
0                0               0                 0                 0

   auto_model_Corolla  auto_model_E400  auto_model_Escape  auto_model_F150  \
0                   0                0                  0                0

   auto_model_Forrestor  auto_model_Fusion  auto_model_Grand Cherokee  \
0                     0                  0                          0
```

```
   auto_model_Highlander  auto_model_Impreza  auto_model_Jetta  \
0                      0                   0                 0

   auto_model_Legacy  auto_model_M5  auto_model_MDX  auto_model_ML350  \
0                  0             0               0                 0

   auto_model_Malibu  auto_model_Maxima  auto_model_Neon  auto_model_Passat  \
0                  0                   0                0                  0

   auto_model_Pathfinder  auto_model_RAM  auto_model_RSX  \
0                      0               0               0

   auto_model_Silverado  auto_model_TL  auto_model_Tahoe  auto_model_Ultima  \
0                     0              0                 0                  0

   auto_model_Wrangler  auto_model_X5  auto_model_X6  csl_per_person_100  \
0                    0              0               0                   0

   csl_per_person_250  csl_per_person_500  csl_per_accident_1000  \
0                   1                   0                      0

   csl_per_accident_300  csl_per_accident_500  \
0                     0                     1

   incident_period_of_day_afternoon  incident_period_of_day_early_morning  \
0                                 0                                      0

   incident_period_of_day_evening  incident_period_of_day_fore-noon  \
0                               1                                 0

   incident_period_of_day_morning  incident_period_of_day_night  \
0                               0                              0

   incident_period_of_day_past_midnight  property_damage  \
0                                     0               0

   police_report_available  collision_en  months_as_customer  age  \
0                        1             1                   5   37

   policy_deductable  policy_annual_premium  umbrella_limit  capital.gains  \
0                500                1145.28             0.0          54735

   capital.loss  number_of_vehicles_involved  bodily_injuries  witnesses  \
0         88553                           1                2          1

   total_claim_amount  injury_claim  property_claim  vehicle_claim  \
0             96200.0          3000             500          58870
```

```
   vehicle_age
0           21
```

[ ]:
```
x = X.drop([
       'vehicle_claim',
        'injury_claim',
        'age',
        'csl_per_accident_500',
        'csl_per_accident_1000',
        'auto_model_Wrangler',
        'insured_sex_MALE',
        'csl_per_accident_300',
        'property_claim',
        'number_of_vehicles_involved'], axis=1)

x.head(1)
```

[ ]:
```
   policy_state_IL  policy_state_IN  policy_state_OH  insured_sex_FEMALE  \
0                0                1                0                   1

   insured_education_level_Associate  insured_education_level_College  \
0                                  1                                0

   insured_education_level_High School  insured_education_level_JD  \
0                                    0                           0

   insured_education_level_MD  insured_education_level_Masters  \
0                           0                                0

   insured_education_level_PhD  insured_occupation_adm-clerical  \
0                            0                                 0

   insured_occupation_armed-forces  insured_occupation_craft-repair  \
0                                0                                 0

   insured_occupation_exec-managerial  insured_occupation_farming-fishing  \
0                                   0                                    0

   insured_occupation_handlers-cleaners  insured_occupation_machine-op-inspct  \
0                                     0                                      0

   insured_occupation_other-service  insured_occupation_priv-house-serv  \
0                                 0                                   1

   insured_occupation_prof-specialty  insured_occupation_protective-serv  \
0                                  0                                   0
```

```
   insured_occupation_sales  insured_occupation_tech-support  \
0                         0                                0

   insured_occupation_transport-moving  insured_hobbies_base-jumping  \
0                                    0                              0

   insured_hobbies_basketball  insured_hobbies_board-games  \
0                           0                            0

   insured_hobbies_bungie-jumping  insured_hobbies_camping  \
0                               0                        0

   insured_hobbies_chess  insured_hobbies_cross-fit  insured_hobbies_dancing  \
0                      0                          0                        0

   insured_hobbies_exercise  insured_hobbies_golf  insured_hobbies_hiking  \
0                         0                    0                       0

   insured_hobbies_kayaking  insured_hobbies_movies  \
0                         0                        1

   insured_hobbies_paintball  insured_hobbies_polo  insured_hobbies_reading  \
0                          0                    0                        0

   insured_hobbies_skydiving  insured_hobbies_sleeping  \
0                          0                         0

   insured_hobbies_video-games  insured_hobbies_yachting  \
0                            0                         0

   insured_relationship_husband  insured_relationship_not-in-family  \
0                             1                                   0

   insured_relationship_other-relative  insured_relationship_own-child  \
0                                    0                               0

   insured_relationship_unmarried  insured_relationship_wife  \
0                               0                          0

   incident_type_Multi-vehicle Collision  incident_type_Parked Car  \
0                                      0                        0

   incident_type_Single Vehicle Collision  incident_type_Vehicle Theft  \
0                                       1                            0

   incident_severity_Major Damage  incident_severity_Minor Damage  \
```

```
0                              0                                  1

   incident_severity_Total Loss  incident_severity_Trivial Damage  \
0                             0                                  0

   authorities_contacted_Ambulance  authorities_contacted_Fire  \
0                                0                           0

   authorities_contacted_None  authorities_contacted_Other  \
0                           0                            1

   authorities_contacted_Police  incident_state_NC  incident_state_NY  \
0                             0                  0                  1

   incident_state_OH  incident_state_PA  incident_state_SC  incident_state_VA  \
0                  0                  0                  0                  0

   incident_state_WV  incident_city_Arlington  incident_city_Columbus  \
0                  0                        0                       0

   incident_city_Hillsdale  incident_city_Northbend  incident_city_Northbrook  \
0                        1                        0                         0

   incident_city_Riverwood  incident_city_Springfield  auto_make_Accura  \
0                        0                           0                 0

   auto_make_Audi  auto_make_BMW  auto_make_Chevrolet  auto_make_Dodge  \
0               0              0                    0                0

   auto_make_Ford  auto_make_Honda  auto_make_Jeep  auto_make_Mercedes  \
0               0                0               0                   0

   auto_make_Nissan  auto_make_Saab  auto_make_Suburu  auto_make_Toyota  \
0                 0               1                 0                 0

   auto_make_Volkswagen  auto_model_3 Series  auto_model_92x  auto_model_93  \
0                     0                    0               1              0

   auto_model_95  auto_model_A3  auto_model_A5  auto_model_Accord  \
0              0              0              0                  0

   auto_model_C300  auto_model_CRV  auto_model_Camry  auto_model_Civic  \
0                0               0                 0                 0

   auto_model_Corolla  auto_model_E400  auto_model_Escape  auto_model_F150  \
0                   0                0                  0                0
```

```
   auto_model_Forrestor  auto_model_Fusion  auto_model_Grand Cherokee  \
0                     0                  0                          0

   auto_model_Highlander  auto_model_Impreza  auto_model_Jetta  \
0                      0                   0                 0

   auto_model_Legacy  auto_model_M5  auto_model_MDX  auto_model_ML350  \
0                  0              0               0                 0

   auto_model_Malibu  auto_model_Maxima  auto_model_Neon  auto_model_Passat  \
0                  0                  0                0                  0

   auto_model_Pathfinder  auto_model_RAM  auto_model_RSX  \
0                      0               0               0

   auto_model_Silverado  auto_model_TL  auto_model_Tahoe  auto_model_Ultima  \
0                     0              0                 0                  0

   auto_model_X5  auto_model_X6  csl_per_person_100  csl_per_person_250  \
0              0              0                   0                   1

   csl_per_person_500  incident_period_of_day_afternoon  \
0                   0                                 0

   incident_period_of_day_early_morning  incident_period_of_day_evening  \
0                                     0                               1

   incident_period_of_day_fore-noon  incident_period_of_day_morning  \
0                                 0                               0

   incident_period_of_day_night  incident_period_of_day_past_midnight  \
0                             0                                      0

   property_damage  police_report_available  collision_en  months_as_customer  \
0                0                        1             1                   5

   policy_deductable  policy_annual_premium  umbrella_limit  capital.gains  \
0                500                1145.28             0.0          54735

   capital.loss  bodily_injuries  witnesses  total_claim_amount  vehicle_age
0         88553                2          1             96200.0           21
```

[ ]:
```
x_train, x_test, y_train, y_test = train_test_split(x, y, train_size=0.8,␣
↪random_state=7)
print('length of X_train and X_test: ', len(x_train), len(x_test))
print('length of y_train and y_test: ', len(y_train), len(y_test))
```

```
length of X_train and X_test:  8168 2043
length of y_train and y_test:  8168 2043
```

[ ]:
```
a_train_scaled = scaler.fit_transform(x_train)
a_test_scaled = scaler.transform(x_test)
```

[ ]:
```
xgb = XGBClassifier()
logreg= LogisticRegressionCV(solver='lbfgs', cv=10)

# prepare configuration for cross validation test harness
seed = 7
# prepare models
models = []
models.append(('LR', LogisticRegressionCV(solver='lbfgs', max_iter=5000,␣
 ↪cv=10)))
models.append(('XGB', XGBClassifier()))

# evaluate each model in turn
results = []
names = []
scoring = 'accuracy'
for name, model in models:
    kfold = model_selection.KFold(n_splits=10, random_state=seed)
    cv_results = model_selection.cross_val_score(model, a_train_scaled,␣
 ↪y_train, cv=kfold, scoring=scoring)
    results.append(cv_results)
    names.append(name)
    msg = "%s: %f (%f)" % (name, cv_results.mean(), cv_results.std())
    print(msg)
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


LR: 0.880265 (0.015103)
/usr/local/lib/python3.6/dist-packages/sklearn/model_selection/_split.py:296:
FutureWarning:

Setting a random_state has no effect since shuffle is False. This will raise an
error in 0.24. You should leave random_state to its default (None), or set
shuffle=True.


XGB: 0.896795 (0.015345)
```

[ ]: